

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Matic Novak

**Integracija plačilnega sistema s  
portalom za izdajo licenc**

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM  
PRVE STOPNJE  
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Dejan Lavbič

Ljubljana, 2016

*Besedilo je oblikovano z urejevalnikom besedil  $\text{\LaTeX}$ .*

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:  
Integracija plačilnega sistema s portalom za izdajo licenc

Tematika naloge:

Večina projektov programskega inženirstva se v zadnjem času osredotoča predvsem na integracijo informacijskih sistemov, z namenom avtomatizacije poslovnih procesov. Na področju elektronskega poslovanje raste uporaba plačilnih sistemov pri informacijskih rešitvah, saj močno izboljšajo uporabniško izkušnjo. V okviru diplomske naloge raziščite področje spletnih plačilnih sistemov, jih kritično ovrednotite in na podlagi prototipa implementirajte plačilni sistem v obstoječo spletno trgovino. Pri integraciji predlagajte prenovno poslovnega procesa, ki ga zahteva uvedba avtomatiziranega plačilnega sistema. Pri implementaciji prototipa obravnavajte tudi varnostna tveganja, povezana s spletnim plačevanjem.



*Zahvaljujem se vsem, ki so kakor koli pripomogli k nastanku te diplomske naloge. Najlepša hvala podjetju, ki mi je omogočilo, da sem se v okviru diplomske naloge soočil z realnim problemom iz prakse in ga poskušal rešiti tudi pod njihovim mentorstvom. Hvala mentorju doc. dr. Dejanu Lavbiču za strokovno pomoč in podporo, predvsem za njegovo neverjetno odzivnost in odlične nasvete. Zahvalil bi se sestri za pomoč, vzpodbudo in podporo. Iskrena zahvala staršem, ki so mi omogočili študij. Zahvala mojemu dekletu za vlivanje poguma in upanja v trenutkih, ko sem to najbolj potreboval. Zahvala gre tudi vsem sošolcem, sošolkam in prijateljem, ki so mi v letih študija pomagali na kakršen koli način ter prispevali k veselim, zabavnim in nepozabnim trenutkom.*

*Hvala!*



Družini.





# Kazalo

Povzetek

Abstract

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Spletni plačilni sistemi</b>	<b>5</b>
2.1	Načini plačila . . . . .	7
2.2	Zunanji plačilni sistemi . . . . .	10
2.3	Razumevanje terminologije plačilnih sistemov . . . . .	10
2.4	Postopek spletne transakcije . . . . .	12
2.5	Slovar ostalih pojmov . . . . .	14
<b>3</b>	<b>Analiza obstoječih plačilnih sistemov</b>	<b>15</b>
3.1	Izbira ustreznega plačilnega sistema . . . . .	15
3.2	Opredelitev odločitvenega modela . . . . .	16
3.3	Funkcija koristnosti . . . . .	18
3.4	Vrednotenje kandidatov . . . . .	21
3.5	Analiza rezultatov odločitvenega procesa . . . . .	22
3.6	Analiza kandidatov . . . . .	24
3.7	Ponudniki preverjanja davčne številke . . . . .	26
<b>4</b>	<b>Pregled tehnologij in orodij</b>	<b>29</b>
4.1	Microsoft Visual Studio . . . . .	29
4.2	Programski jezik C# . . . . .	30

4.3	ASP.NET MVC - WebApi . . . . .	31
4.4	Microsoft SQL Server . . . . .	32
<b>5</b>	<b>Platforma za izdajo licenc</b>	<b>33</b>
5.1	Določitev prodajnega partnerja . . . . .	33
5.2	Konfiguracija produktov za prodajo . . . . .	34
5.3	Prednastavljanje obvestilnih sporočil . . . . .	37
5.4	Pregled spletnih nakupov . . . . .	38
5.5	Pregled zahtevkov z REST API-ja . . . . .	39
<b>6</b>	<b>Avtomatizacija izdajanja licenc</b>	<b>41</b>
6.1	Spletne storitve . . . . .	41
6.2	Prenova poslovnega procesa . . . . .	42
6.3	Identifikacija produktov . . . . .	45
6.4	Potek naročila . . . . .	47
6.5	Rezultat avtomatizacije . . . . .	53
<b>7</b>	<b>Skrb za varnost</b>	<b>55</b>
7.1	Zavarovana povezava HTTPS . . . . .	56
7.2	Enosmerna zgoščevalna funkcija MD5 . . . . .	57
<b>8</b>	<b>Sklepne ugotovitve</b>	<b>59</b>
	<b>Literatura</b>	<b>62</b>



# Seznam uporabljenih kratic

kratica	angleško	slovensko
<b>HTTP</b>	Hypertext Transfer Protocol	protokol za prenos hiperteksta
<b>URL</b>	Uniform Resource Locator	internetni naslov, na katerem se nahaja vsebina
<b>URI</b>	Uniform Resource Identifier	enolični krajevniki vira
<b>SOAP</b>	Simple Object Access Protocol	protokol za spletne storitve, ki temelji na XML
<b>REST</b>	Representational State Transfer	predstavitveni prenos stanj
<b>API</b>	Application Programming Interface	aplikacijski programski vmesnik
<b>XML</b>	Extensible Markup Language	razširljiv označevalni jezik
<b>JSON</b>	JavaScript Object Notation	preprost podatkovni format za izmenjavo
<b>WSDL</b>	Web Services Description Language	standardizirani protokol za opis storitev, ki jih nudi sistem
<b>UDDI</b>	Universal Description, Discovery and Integration	sistem za oglaševanje storitev, ki jih nudi določena aplikacija
<b>FTP</b>	File Transfer Protocol	protokol za prenos datotek
<b>SMTP</b>	Simple Mail Transport Protocol	internetni protokol za prenos elektronske pošte
<b>TLS</b>	Transport Layer Security	varnost prenosnega sloja
<b>SSL</b>	Secure Socket Layer	sloj varnih vtičnic
<b>TCP</b>	Transmission Control Protocol	protokol s kontrolo prenosa

# Povzetek

**Naslov:** Integracija plačilnega sistema s portalom za izdajo licenc

**Avtor:** Matic Novak

Sodobni informacijski sistemi stremijo k medsebojni integraciji, rezultat le-te pa so avtomatizirani procesi. V poslovnem svetu odražajo boljšo učinkovitost znotraj organizacij, strankam pa omogočajo boljšo uporabniško izkušnjo. Namen diplomskega dela je implementacija komunikacije med portalom za izdajo licenc in spletnim plačilnim sistemom. Tako se bomo znebili nepotrebne čakanja na administrativnega delavca, ki ročno izdaja licence, hkrati pa bomo preprečili napake pri ročnem vnosu. V prvem delu bomo analizirali spletne plačilne sisteme in predstavili odločitveni model, s pomočjo katerega smo izbrali najustreznejšega. Podrobneje bomo predstavili portal za izdajo licenc, kjer se beležijo vsi podatki o prodajalcih, kupcih, naročilih in licenčnih številkah. V osrednjem delu bomo prikazali prenovo obstoječega poslovnega procesa in implementacijo komunikacije med obema sistemoma. Skozi celoten proces izdaje licenčnih številčk imamo opravka z zaupnimi podatki, zato na koncu sledi poglavje o tem, kako smo poskrbeli za zaščito in varnost.

**Ključne besede:** integracija, prenova poslovnega procesa, plačilni sistem, licenčna številka, spletna storitev, programska oprema, varnost.



# Abstract

**Title:** The integration of the payment system and a license issuing platform

**Author:** Matic Novak

Modern information systems aim to mutual integration which results in automated processes. In the business world they reflect better effectiveness inside the organizations and enable clients to have better user experience. The goal of this thesis is the implementation of communication between the license issuing platform and online payment system. These processes help to get rid of typos and the annoying wait for issuing the licences on hand. In the first part we will analyze online payment systems and present the model that we used for choosing the most appropriate one. We will present the license issuing platform where all the data about salesmen, buyers, orders and licence numbers is being kept. We will also show business process reengineering and implementation of communication between both of the systems. Through the whole process of issuing licence numbers we have to deal with confidential data that is why there is a chapter about security and safety at the end.

**Keywords:** integration, business process renovation, payment system, license number, web service, software, security.





# Poglavje 1

## Uvod

Nakupovanje preko spleta ponuja veliko prednosti, med drugim udobje našega doma, varčevanje pri porabi goriva in našega časa. Preko spleta lahko danes kupimo praktično vse, vključno z gospodinjskimi aparati, oblačili, mobilnimi aparati, darili, storitvami. Ob spletnih nakupih ne smemo pozabiti na digitalne dobrine, kot so elektronske knjige, programska oprema in njej namenjeni licenčni ključi, oblikovne predloge idr. Kljub temu raziskave kažejo, da se v svetu opravi le do 30 % nakupov preko spleta, ostalih 70 % pa v fizičnih trgovinah [10]. Obstaja veliko razlogov za navedeno porazdelitev:

- kupci se izogibajo včasih nepredvidljivo dolgim dostavnim rokom, izdelek imajo raje nemudoma v svojih rokah;
- večina ljudi si želi izdelek pred nakupom ogledati, ga morda celo preizkusiti;
- nekateri trdijo, da s fizičnimi nakupi podpirajo lokalne prodajalne.

V nadaljnjih izsledkih raziskav lahko največkrat zasledimo, da imajo kupci pomisleke glede varnosti spletnega plačevanja iz odlaganja osebnih podatkov pri spletnih trgovcih. To je pogosto razlog, da željen izdelek poiščejo na spletu, nakup pa opravijo v lokalni trgovini. Vendar pa zaupanje v spletne trgovine in spletne plačilne sisteme skozi leta raste. Elektronsko poslovanje pridobiva na priljubljenosti, saj se vse več ljudi odloča za nakupovanje preko

spleta [10]. Na trgu se je zato pojavila vrsta plačilnih sistemov, ki so bolj ali manj poznani med uporabniki. Tisti z večjo prepoznavnostjo so tudi dobro sprejeti.

Kot že omenjeno, se programska oprema v večji meri prodaja preko spleta. Stranke, ki se odločijo za nakup programov, si želijo nakup opraviti hitro in enostavno. Najboljše spletne trgovine stremijo k izpolnjevanju pričakovanj in potreb strank. Ena izmed teh je gotovo spletno plačevanje in tiste trgovine, ki tega ne omogočajo, niso konkurenčne. Trgovec tako avtomatizira plačila, znebi se ročnega preverjanja nakazil. To zanj pomeni manj dela, za stranko pa hitrejšo izdajo licenčne številke. Iz konteksta lahko razberemo, da imamo opravka z dvema ločenima sistemoma, vsak izmed njiju pa opravlja svoj del naloge.

Na eni strani imamo spletni portal za izdajo licenčnih števil, kamor prodajalec vnese podatke o stranki ter o programskem paketu, ki ga je stranka plačala. Sistem ob uspešnem vnosu podatkov vrne licenčno številko, ki je vezana na izbrani programski paket in vnešeno stranko. Prodajalec licenčno številko posreduje stranki, ta pa lahko začne z uporabo kupljenega programskega paketa. Stranka in prodajalec morata v omenjenem postopku stopiti v kontakt preko elektronske pošte ali telefona, da izpeljeta naročilo.

Na drugi strani imamo spletni plačilni sistem, ki ponuja vrsto storitev. Stranka lahko na zavarovanem portalu izbere programski paket, katerega želi kupiti. Vnese osebne podatke ter podatke za spletno plačilo. Plačilni sistem vnešene podatke preveri pri banki. V kolikor banka odobri transakcijo, plačilni sistem stranki izda serijsko številko in ta lahko začne z uporabo ravnokar kupljenega programskega paketa. V tem primeru stranka ni prisiljena komunicirati s prodajalcem, saj lahko celoten postopek opravi sama. Stranka tako hitreje pride do željenega izdelka ali storitve, prodajalec pa ima s prodajo manj dela in stroškov.

Cilj diplomske naloge je razviti prototip rešitve, ki bo omogočal komunikacijo med portalom za izdajo licenc in spletnim plačilnim sistemom po zavarovani povezavi. Plačilni sistem bo ob strankini zahtevi izvedel klic na portal za izdajo licenc. Klic bo vseboval podatke o stranki in o željenem programskem paketu. Portal za izdajo licenc bo ob uspešni avtorizaciji plačilnega sistema vrnil licenčno številko. Le-to bo plačilni sistem skupaj z računom prikazal stranki na zaslon in ji hkrati vsebino posredoval na elektronski naslov.

V naslednjem poglavju bomo podrobneje predstavili, kaj spletni plačilni sistemi so, terminologijo in način izbire ustreznega plačilnega sistema. Nato se bomo posvetili analizi obstoječih ponudnikov tovrstnih storitev. Sledilo bo poglavje o tehnologijah, ki smo jih uporabili v okviru diplomskega dela za samo implementacijo avtomatizacije izdajanja licenc. V petem poglavju bomo predstavili platformo, ki služi upravljanju s strankami, produkti in z licenčnimi številkami. Za tem pride na vrsto ključno poglavje, kjer si bomo podrobneje ogledali prenovljen poslovni proces izdajanja licenc in implementacijo le-tega. Ko govorimo o spletnih transakcijah, ki vključujejo finančne podatke, je varnost pogost pomislek uporabnikov, zato bomo poglavje namenili tudi tej problematiki. V sklepnem poglavju bomo poskušali oceniti rezultat našega dela in delovanje prototipa.



## Poglavje 2

# Spletni plačilni sistemi

V zadnjih letih smo priča veliki porasti elektronskega poslovanja, ki je ustvarilo potrebe po novih plačilnih metodah. Spletno plačevanje je pomemben dosežek na področju računalništva in financ, ki vzpodbuja potrošnjo. Ideja je v nudenju priročnega načina plačila strankam. Na ta način ne zadovoljimo le potrebe po spletnem plačevanju, temveč tudi ustvarjamo vse več povpraševanja. Vsako leto spletni plačilni trg zraste za 30 % [19].

Plačilni sistem je vsak sistem za poravnavo finančnih transakcij s prenosom denarne vrednosti. Vključuje institucije, instrumente, ljudi, pravila, postopke, standarde in tehnologijo, ki omogočajo takšno izmenjavo [2].

Metodi plačilne storitve in plačilni sistem vključujeta ponudnika plačilne metode, kupca in prodajalca. Kupec se prijavi v storitev in pridobi unikatni identifikator, ki mu omogoča izvajanje vseh transakcij s plačilno storitvijo ponudnika. Kupec komunicira s ponudnikom plačilne storitve in olajša plačila prodajalcem skozi ponudnika plačilne storitve ter ostane anonimen [6].

Po načelu nekaterih uveljavljenih metod plačevanja prek spleta se dogovor o plačevanju vzpostavi pri ponudniku plačilnih storitev. Ta ureja pogoje sodelovanja med prodajalcem in kupcem. Dogovor vključuje povezavo med prodajalčevim in kupčevim računom pri ponudniku plačilne storitve. Po vzpostavitvi dogovora ponudnik plačilne storitve prejme zahtevo za plačilo za določen nakup. Zahteva vključuje unikatno kodo, na podlagi katere je

povezana z dogovorom o plačevanju. Zahteva za plačilo je izvršena, če ustreza pogojem, določenim v dogovoru o plačevanju [18].

Raziskave in študije v evropskih državah [17] so pokazale, da so najmočnejši dejavniki na poti do uspeha določenega spletnega plačilnega sistema tehnološke, ekonomske in človeške narave. Da bi razumeli vlogo posameznih dejavnikov, bomo v nadaljevanju na kratko predstavili rezultate raziskav.

- Glede na odzive uporabnikov je **enostavnost** uporabe najpomembnejši faktor. 96 % ljudi, ki so v preteklosti že imeli izkušnjo s plačevanjem na spletu, je dejalo, da je kreditna kartica najenostavnejša za uporabo.
- **Varnost in zanesljivost** sta naslednja najpomembnejša dejavnika za 98,4 % anketirancev. 75,2 % anketirancev nemudoma preneha z uporabo spletnih plačilnih storitev, če prejme informacijo o kakršnikoli napaki v varnostnem sistemu njihove banke.
- **Zaupanje** v sistem je še en pomemben dejavnik v očeh anketirancev. 97,6 % anketirancev zaupa le v sisteme, ki so bili ustanovljeni in potrjeni s strani vladnih organov.
- 85 % vprašanih se strinja, da mora dobra spletna trgovina ponuditi široko paleto izbire načina plačila. S tem naslavljajo **uporabnost** plačilnega sistema.
- 58,3 % vprašanih ni pripravljenih zaupati vseh svojih osebnih podatkov plačilnemu sistemu. To za anketirance ne pomeni, da mora biti njihova identiteta popolnoma anonimna, vendar pa vztrajajo pri **zasebnosti** na razumni ravni.

## 2.1 Načini plačila

Spletne trgovine ponujajo najrazličnejše načine plačila. Na seznam smo poskušali uvrstiti kar največ opcij, ki jih ponuja trg - od klasičnega gotovinskega plačila, kjer ne govorimo o spletnem plačevanju, pa do najmodernejših načinov plačevanja:

- z gotovino po povzetju,
- po predračunu (nakazilo na transakcijski račun),
- preko spletnih kuponov/darilnih bonov,
- s kreditnimi in debetnimi karticami,
  - VISA, <sup>1</sup>
  - VISA Electron,
  - VISA Debit,
  - MasterCard, <sup>2</sup>
  - MasterCard Debit,
  - Maestro, <sup>3</sup>
  - American Express, <sup>4</sup>
  - Diners Club International, <sup>5</sup>
  - Discover, <sup>6</sup>
  - JCB, <sup>7</sup>
  - UnionPay, <sup>8</sup>

---

<sup>1</sup><https://www.visaeurope.com/>

<sup>2</sup><http://www.mastercard.com/si/consumer/>

<sup>3</sup><http://www.maestrocard.com/si/>

<sup>4</sup><http://www.americanexpress.si/>

<sup>5</sup><http://www.dinersclub.si/>

<sup>6</sup><https://www.discover.com/>

<sup>7</sup><http://www.jcbcard.com/>

<sup>8</sup><http://www.unionpayintl.com/>

- s kripto valutama (najpogosteje Bitcoin),
- preko PayPal sistema, <sup>9</sup>
- preko Monete, <sup>10</sup>
- preko ostalih zunanjih plačilnih servisov,
  - Skrill, <sup>11</sup>
  - Stripe, <sup>12</sup>
  - 2Checkout, <sup>13</sup>
  - FastSpring, <sup>14</sup>
  - BlueSnap, <sup>15</sup>
  - PayLane, <sup>16</sup>
  - Authorize.net, <sup>17</sup>
  - ProPay, <sup>18</sup>
  - Amazon Payments, <sup>19</sup>
  - Google Wallet. <sup>20</sup>

---

<sup>9</sup><https://www.paypal.com/si/webapps/mpp/home>

<sup>10</sup><https://www.moneta.si/>

<sup>11</sup><https://www.skrill.com/en/>

<sup>12</sup><https://stripe.com/>

<sup>13</sup><https://www.2checkout.com/>

<sup>14</sup><http://www.fastspring.com/>

<sup>15</sup><https://home.bluesnap.com/>

<sup>16</sup><http://paylane.com/>

<sup>17</sup><http://www.authorize.net/>

<sup>18</sup><https://www.propay.com/>

<sup>19</sup><https://payments.amazon.com/>

<sup>20</sup><https://www.google.com/wallet/>



Plačilo z gotovino po povzetju je najstarejši način plačevanja v spletnem trgovanju, ki je še danes pogosto uporabljen. S strani trgovcev ni najbolj priljubljen, ker nimajo zagotovila, da bo kupec izdelek zares prevzel in zanj plačal. V primeru, da se kupec v času dostave izdelka odloči, da ga ne potrebuje, paket enostavno zavrne, trgovcu pa s tem povzroči strošek pakiranja in dostave.

V primeru nakupa digitalnih dobrin ne moremo računati na plačilo z gotovino po povzetju. Nihče namreč ne dostavlja elektronskih knjig na dom, prav tako ne licenčnih številčk za programsko opremo. Bilo bi namreč nesmiselno, da kupec v 21. stoletju čaka dan ali celo več, da mu dostavna služba pripelje nekaj, kar si lahko sam preko spleta prenese na svojo elektronsko napravo v nekaj trenutkih. Trgovec tudi v tem primeru za svoje storitve zahteva denar. Kupec nima fizičnega stika s prodajalcem, zato poseže po spletnem plačevanju in spletnih plačilnih sistemih.

Na žalost ima vsak od zgoraj naštetih načinov plačila svoje omejitve. Pri kreditnih karticah je problem ta, da je nima ravno vsak kupec. Tudi delež tistih, ki kreditno kartico imajo, ne zaupa pošiljanju občutljivih podatkov s kartice po internetu. To jih bo odvrnilo od uporabe omenjene metode. Paypal in podobni plačilni sistemi zahtevajo registracijo in uporabljajo ločene uporabniške račune, kar pomeni dodatno porabo časa in neudobje pri spletnem nakupovanju [25].

## 2.2 Zunanji plačilni sistemi

Spletno plačevanje predstavlja velik tehnični in varnostni zalogaj, zato spletne trgovine to odgovornost po navadi prelagajo na tretjo osebo - spletni plačilni servis. Ti poleg prenosa denarja med kupčevim in trgovčevim računom implementirajo tudi visoko stopnjo varnosti in zaupnosti. Vsaka storitev oziroma funkcionalnost pomeni strošek za spletno trgovino. Ponudniki plačilnih storitev v zameno za uporabo zaračunavajo provizijo za vsako transakcijo. Nekateri poleg tega zaračunavajo tudi vzpostavitveno pristojbino in/ali mesečno naročnino. Spletni plačilni servisi nas tako prikrajšajo za določen odstotek zneska celotnega nakupa, vendar nam omogočajo sodobne oblike plačevanja in zadovoljstvo kupcev. V enem izmed naslednjih poglavji bomo med sabo primerjali različne plačilne sisteme, tako po funkcionalnostih kot tudi po provizijah, ki jih moramo odšteti za uporabo.

## 2.3 Razumevanje terminologije plačilnih sistemov

Obstaja več izrazov, ki se večkrat uporabljajo pri opisovanju spletnega plačila [3]:

- plačilni prehod (angl. payment gateway),
- plačilni procesor (angl. payment processor),
- ponudnik plačilnih storitev (angl. payment provider),
- plačilni servis oziroma plačilni sistem (angl. payment service or payment system),
- trgovalni račun (angl. merchant account).

Čeprav so pojmi različni, se vsi sklicujejo na družbo (podjetje), storitev ali aplikacijo, ki deluje kot finančni posrednik med našo spletno stranjo in stranko ter med bančnima računoma. Vsak skrbi za eno izmed komponent spletnih transakcij in obdelavo spletnih plačil.

**Plačilni prehod** je storitev, ki prejme zahtevo za spletno plačilo z naše spletne strani in jo usmeri k plačilnemu procesorju.

**Plačilni procesor** je storitev, ki validira kupčeve podatke o kreditni kartici in preveri, če ima dovolj sredstev na svojem računu za kritje plačila. Če je sredstev dovolj, je transakcija avtorizirana in denar se prenese s kupčevega računa. Status transakcije se posreduje plačilnemu prehodu, ta pa ga posreduje spletni strani prodajalca.

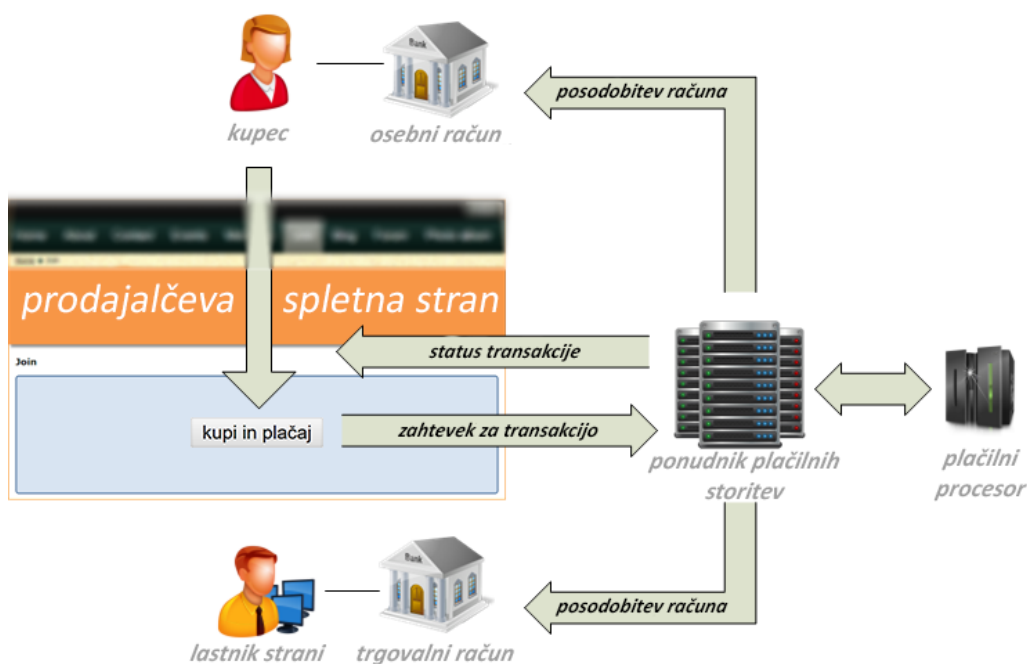
**Ponudnik plačilnih storitev** je podjetje, ki opravlja storitve plačilnega prehoda ali plačilnega procesorja. V nekaterih primerih obe storitvi opravlja eno samo podjetje.

**Plačilni servis oziroma plačilni sistem:** V primeru, da ponudnik plačilnih storitev ponuja različne tipe plačilnih prehodov z različnimi značilnostmi in različnimi cenovnimi modeli, je vsaka izmed teh možnosti plačilni servis oziroma plačilni sistem (npr. PayPal je ponudnik plačilnih storitev, ki ponuja vrsto le-teh: PayPal Payflow Pro, PayPal Express Checkout ...).

**Trgovalni račun:** V trenutku, ko spletna transakcija uspe, se denar prenese s kupčevega računa na trgovalni račun trgovca. To je posebna vrsta bančnega računa, ki se uporablja izključno za vodenje sredstev, prejetih v transakcijah s kreditnimi in debetnimi karticami. Za sprejemanje spletnih transakcij mora trgovec odpreti račun pri ponudniku plačilnih storitev. Prejeta sredstva se nabirajo na trgovalnem računu, ki pa se redno prenašajo na bančni račun podjetja.

## 2.4 Postopek spletne transakcije

Da bomo v celoti razumeli, kako potekajo spletna plačila, sledimo transakciji od začetka do konca. Na sliki 2.1 je predstavljeno, v kakšni interakciji delujejo naša spletna stran, kupec in plačilni sistem [3].



Slika 2.1: Postopek spletne transakcije

Če želimo začeti s spletnimi plačili, običajno potrebujemo:

- trgovalni račun (nekateri plačilni sistemi ne zahtevajo trgovalnega računa, npr. PayPal);
- račun pri ponudniku plačilnih storitev;
- spletno stran z gumbom „kupi“, ki sproži postopek transakcije.

Ko vzpostavimo našo spletno trgovino in jo povežemo s plačilnim servisom, bodo lahko obiskovalci preko spleta plačali za naše produkte. Spletni

plačilni postopek se začne, ko obiskovalec klikne gumb, da želi za produkte iz naše trgovine plačati preko spleta.

V obrazec, ki se pojavi, kupec vnese podatke o svoji kreditni kartici in pošlje zahtevo za transakcijo. Odvisno od našega spletnega ponudnika plačilnih storitev, se lahko obrazec prikaže na naši spletni strani, ali pa je kupec preusmerjen na obrazec, ki se nahaja na spletni strani našega ponudnika plačilnih storitev. O prednostih in slabostih obojega bomo še spregovorili v nadaljevanju.

Zahteva za transakcijo, skupaj s podatki o kreditni kartici, se varno prenese do plačilnega prehoda, s katerim upravlja naš ponudnik plačilnih storitev. Podatki so šifrirani, zato nihče, vključno z nami, ne more pregledovati osebnih in finančnih podatkov kupca.

Naš ponudnik plačilnih storitev uporabi varno storitev obdelave plačila - bodisi svojo, bodisi storitev kakšnega drugega ponudnika. V obdelavi preveri podatke o kreditni kartici in, ali ima kupec dovolj sredstev za uspeh transakcije.

Če so prijavitni podatki kupca veljavni in je na voljo dovolj sredstev za dokončanje transakcije, ponudnik plačilnega sistema sproži prenos sredstev iz kupčevega bančnega računa na trgovčev trgovalni račun in obvesti našo spletno stran, da je bila transakcija uspešna. Te informacije lahko uporabimo za samodejno posodobitev zapisov (na spletni strani, v podatkovni bazi...). Če je transakcija zavrnjena iz kakršnega koli razloga (neveljavni prijavitni podatki ali pomanjkanje denarja na računu), se sredstva ne prenesejo. Plačilni sistem našo spletno stran obvesti o napaki [3].

Potreben je razmislek o lokaciji obrazca za vnos plačilnih podatkov. Kot že omenjeno, se le ta lahko nahaja na naši spletni strani. Prednosti tega načina so, da stranke v postopku nakupa licence ne preusmerjamo drugam in ostane na prodajalčevi spletni strani, kar daje občutek varnosti, stranka se počuti bolj lagodno. Še ena prednost je v tem, da lahko obliko obrazca popolnoma prilagodimo svojim željam in potrebam. Po drugi strani je „slabost“

ta, da moramo sami poskrbeti za varnost pri pošiljanju podatkov in komuniciranju s plačilnim servisom. Skoraj nujno je, da spletna stran uporablja protokol HTTPS.

Druga možnost pa je, da se obrazec za vnos plačilnih podatkov nahaja na spletni strani našega ponudnika plačilnih storitev. Prednost je prav gotovo ta, da nam ni potrebno skrbeti za varnost. Nekatere stranke se tako počutijo bolj varno in imajo večjo mero zaupanja, če izbrani plačilni servis poznajo in so ga že uporabljale. Slabost je, da izgleda obrazca običajno ne moremo dobro prilagajati našim željam, v nekaterih primerih se njegove oblike praktično sploh ne da urejati. Poleg tega je stranka preusmerjena na drugo spletno stran, kar lahko povzroči zmedo ali vzbudi dvome.

## 2.5 Slovar ostalih pojmov

**Debetna kartica** je plačilna kartica, ki imetniku omogoča negotovinsko plačevanje in dvige gotovine, ob tem pa se njegov transakcijski račun bremeni sproti. Ob odprtju osebnega računa najpogosteje dobimo debetno kartico za njegovo uporabo [20].

**Kreditna kartica** je plačilna kartica, ki imetniku omogoča negotovinsko plačevanje in dvige gotovine v okviru zneska kredita, ki ga ima imetnik odobrenega na kartici. Znesek porabljenega kredita se odplačuje v obrokih na vnaprej izbrani dan v mesecu (8., 18. ali 28.) [20].

**Razlika med debetnimi in kreditnimi karticami:** Pri nakupovanju ali dvigovanju gotovine z debetno kartico se znesek takoj odšteje z našega računa, pri uporabi kreditne kartice pa prejmemo mesečni obračun. Če potrebnega zneska na dogovorjeni datum ne odplačamo ali če dvignemo gotovino, se lahko pojavijo dodatni stroški [15].

## Poglavje 3

# Analiza obstoječih plačilnih sistemov

### 3.1 Izbira ustreznega plačilnega sistema

Obstaja množica ponudnikov plačilnih sistemov, vsak z drugačnim poudarkom in naborom funkcionalnosti. Eni so usmerjeni k manjšim organizacijam, drugi k večjim, tretji k neprofitnim ... Večina jih ponuja dodatne storitve, na primer zaščito proti goljufijam, vendar zanje tudi zaračunavajo [3].

Seznam nekaterih dejavnikov, ki jih moramo upoštevati, ko začnemo s primerjavo ponudnikov plačilnih storitev:

- Podpirajo valute, ki jih želimo uporabljati pri poslovanju?
- Kakšne provizije zaračunavajo (za transakcijo, mesečne, vzpostavitvene)?
- Kakšen obseg prometa pričakujete na vaši spletni strani?
- Se plačila obdelujejo na vaši spletni strani ali na strani ponudnika plačilne storitve? Vam je vseeno?
- Katere vrste plačil sprejemajo - kreditne kartice, debetne kartice, PayPal?
- Podpirajo ponavljajoča plačila (npr. za naročnine)?

Želeli smo poiskati plačilni servis, ki bi se kar najbolj približal našim zahtevam, zato smo pregledali vse opaznejše predstavnike na tem področju. Imeli smo čisto splošne zahteve, ki vključujejo nizek strošek samih transakcij ter enostavnost integracije obeh sistemov. Bolj kot sta sistema med sabo kompatibilna, hitreje steče integracija, komunikacija med njima pa je enostavnejša. Posledično je tudi strošek integracije nižji. Izbrani ponudniki, ki smo jih vključili v primerjavo, imajo skupaj večinski tržni delež. Potrebujemo zanesljiv sistem, ki nam ga uveljavljeni ponudniki prav gotovo lahko zagotovijo, zato nas manjši niti niso zanimali.

## 3.2 Opredelitev odločitvenega modela

Po pregledu področja smo povzeli funkcionalnosti, ki jih obstoječi plačilni sistemi ponujajo. Med izbiro kriterijev se nismo želeli preveč omejevat, zato smo vključili le nekaj ključnih, ki zadovoljujejo zgoraj omenjene zahteve. S pridobljenim znanjem smo izbrali tri, ki so odločali o končni izbiri plačilnega sistema:

- **strošek transakcije:** Večina ponudnikov zaračunava provizijo v dveh delih - prvi je statičen, ne glede na znesek transakcije, drugi del pa je izražen v procentih. Vsota obeh delov predstavlja strošek posamezne transakcije. Vsakemu kandidatu smo določili vrednost omenjenega kriterija z zalogo vrednosti: visok, srednji, nizek. Želeli smo najti plačilni servis z najnižjo provizijo.
- **ponujane funkcionalnosti:** Sledi kriterij za ocenjevanje ponujenih funkcionalnosti. Nekateri plačilni servisi jih ponujajo več, drugi manj. Osredotočili smo se zgolj na dve, ki sta se nam zdeli precej pomembni: preverjanje davčne številke in konfiguracija produktov. Zaloga vrednosti je sestavljena iz elementov: vse, ena, ena s pomočjo, nobena. Zapisane so v padajočem vrstnem redu. Najboljši kandidat je tisti, ki ponuja vse, najslabši pa tisti, ki ne ponuja nobene funkcionalnosti.



- **preverjanje davčne številke:** Podjetja svoje programe ponujajo na mednarodnem trgu, kjer se davek na dodano vrednost obračunava po različnih stopnjah. Nekateri plačilni servisi ponujajo storitev, ki validira strankin vnos davčne številke. V bazi podatkov preveri, če davčna številka zares obstaja, je v danem trenutku aktivna in iz katere države prihaja. Glede na ta podatek stranki zaračuna ustrezno stopnjo davka. Vse več podjetij se orientira na mednarodni trg, zato smo pri izbiri plačilnega sistema zahtevali, da ponuja omenjeno funkcionalnost. Zaloga vrednosti tega kriterija vsebuje: ne, zunanja storitev, da. Vrednosti so našteje od najslabše do najboljše, torej, v najboljšem primeru plačilni servis ponuja preverjanje davčne številke kot vgrajeno funkcionalnost.
- **konfiguracija produktov:** Tu lahko plačilne servise zopet delimo na dva dela. Eni zahtevajo, da produkte, ki jih podjetje želi prodajati, v celoti konfiguriramo v nadzorni plošči samega plačilnega servisa. To vključuje vnos imena produkta, opisa produkta, konfiguracijo cene, ki je lahko različna glede na lokalizacijo stranke itd. Spet drugi omogočajo, da vse našteje podatke „prinesemo s sabo” s klikom na ustrezno povezavo v prodajalčevi spletni trgovini. Tam se mora nahajati ustrezen obrazec (angl. form). Klik povezave v bistvu pomeni pošiljanje podatkov s tega obrazca k plačilnemu servisu po metodi HTTP Post. Prodajalec podatke, ki se nahajajo v obrazcu, vnaprej pripravi. To (po navadi) stori z ustreznim klicem na REST API ponudnika plačilnega servisa. Klic vsebuje podobne podatke, kot smo jih našteji zgoraj, le, da so formatirani v ustrezni obliki, ki jo predpiše ponudnik plačilnega servisa. REST API prodajalcu vrne kriptiran žeton, ki enolično določa prodajalčev produkt. Kupec s klikom dejansko pošlje ta žeton in tako prodajalec in plačilni servis vesta, kateri produkt želi kupec kupiti.  
Zaloga vrednosti je v tem primeru manjša. Zanimalo nas je le, če

je produkte potrebno vnaprej konfigurirati ali ne. Kandidati, ki ne zahtevajo vnaprejšnje konfiguracije produktov, so v tem pogledu boljši. Glede na to, da podjetje uporablja platformo za izdajo licenc, kjer so produkti že v celoti konfigurirani, smo sprva preferirali plačilne servise, kjer produktov ni potrebno konfigurirati v njihovi nadzorni plošči. Med končnim izborom ponudnika smo se zaradi različnih interesov temu uklonili.

- **zahtevnost integracije:** Preveriti je potrebno, kakšen način integracije ponujajo plačilni servisi in v kolikšni meri je obstoječa platforma za izdajo licenc že pripravljena za integracijo z vsakim izmed teh. Časovna zahtevnost pomembno vpliva na strošek celotnega projekta. Zaloga vrednosti slednjega kriterija je sestavljena iz: visoka, srednja, nizka. Vrednosti so zopet našteje od najslabše do najboljše. Visoka zahtevnost predstavlja najslabši rezultat, nizka pa najboljšega.

### 3.3 Funkcija koristnosti

Funkcije koristnosti so pri večkriterijskem odločanju v obliki odločitvenih pravil, po katerih se vrednosti posameznih parametrov združujejo v skupno spremenljivko. Določene so v obliki tabel za vsak izpeljani kriterij po principu kaj-če. Določimo jih lahko le za attribute, ki niso listi [4].

V našem primeru smo morali definirati dve funkciji koristnosti. Prvo smo določili za atribut „ponujane funkcionalnosti”, ki je izpeljan iz dveh podatributov: preverjanje davčne številke in konfiguracija produktov. Tabela 3.1 prikazuje odločitvena pravila, po katerih smo izpeljanemu atributu priredili vrednosti.

preverjanje davčne številke	konfiguracija produktov	ponujane funkcionalnosti
ne	zahtevana	nobena
ne	nezahtevana	ena
zunanja storitev	zahtevana	ena s pomočjo
zunanja storitev	nezahtevana	ena
da	zahtevana	ena
da	nezahtevana	vse

Tabela 3.1: Odločitvena pravila za atribut ponujane funkcionalnosti

Drugo funkcijo koristnosti smo morali določiti za korenski atribut odločitvenega modela. Ta temelji na treh kriterijih, katere smo podrobno opisali v poglavju 3.2. Kandidate smo glede na vrednosti omenjenih kriterijev uvrstili v tri razrede: najustreznejši, srednje ustrezen, neustrezen. Odločitvena pravila, po katerih smo razvrščali kandidate, so na voljo v tabeli 3.2.

strošek transakcije	ponujane funkcionalnosti	zahtevnost integracije	izbira plačilnega sistema
visok	vse	visoka	srednje ustrezen
visok	vse	srednja	srednje ustrezen
visok	vse	nizka	najustreznejši
visok	ena	visoka	srednje ustrezen
visok	ena	srednja	srednje ustrezen
visok	ena	nizka	najustreznejši
visok	ena s pomočjo	visoka	neustrezen
visok	ena s pomočjo	srednja	neustrezen
visok	ena s pomočjo	nizka	neustrezen
visok	nobena	visoka	neustrezen
visok	nobena	srednja	neustrezen

visok	nobena	nizka	neustrezen
srednji	vse	visoka	srednje ustrezen
srednji	vse	srednja	najustreznejši
srednji	vse	nizka	najustreznejši
srednji	ena	visoka	srednje ustrezen
srednji	ena	srednja	srednje ustrezen
srednji	ena	nizka	najustreznejši
srednji	ena s pomočjo	visoka	neustrezen
srednji	ena s pomočjo	srednja	neustrezen
srednji	ena s pomočjo	nizka	neustrezen
srednji	nobena	visoka	neustrezen
srednji	nobena	srednja	neustrezen
srednji	nobena	nizka	neustrezen
nizek	vse	visoka	srednje ustrezen
nizek	vse	srednja	najustreznejši
nizek	vse	nizka	najustreznejši
nizek	ena	visoka	srednje ustrezen
nizek	ena	srednja	srednje ustrezen
nizek	ena	nizka	najustreznejši
nizek	ena s pomočjo	visoka	neustrezen
nizek	ena s pomočjo	srednja	neustrezen
nizek	ena s pomočjo	nizka	neustrezen
nizek	nobena	visoka	neustrezen
nizek	nobena	srednja	neustrezen
nizek	nobena	nizka	neustrezen

Tabela 3.2: Odločitvena pravila za korenski atribut

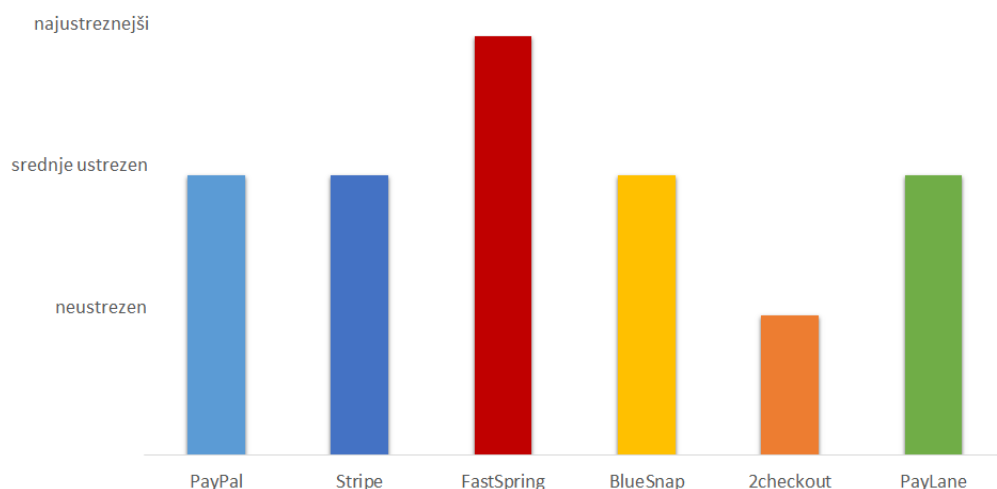
3.4 Vrednotenje kandidatov

ponudnik	strošek transakcije		ponujane funkcionalnosti			zahtevnost integracije
	odstotek [%]	fiksno [\$]	vrednost	preverjanje davčne številke	konfiguracija produktov	vrednost
PayPal	3,40	0,30	srednji	zunanja storitev	nezahtevana	ena visoka
Stripe	2,90	0,30	nizek	zunanja storitev	nezahtevana	ena visoka
FastSpring	8,90	0	visok	da	zahtevana	ena nizka
	5,90	0,95				
BlueSnap	3,90	0,30	visok	da	nezahtevana	vse visoka
2checkout	2,40	0,30	nizek	ne	zahtevana	nobena visoka
PayLane	2,80	0,25	nizek	ne	nezahtevana	ena visoka

Tabela 3.3: Primerjava plačilnih servisov

### 3.5 Analiza rezultatov odločitvenega procesa

Pri odločanju nam je pomagal program DEXi [4], s pomočjo katerega smo zgradili večkriterijski odločitveni model. Vnesli smo zgoraj omenjene kriterije in nastavili pravila. Izmed šestih kandidatov nam je program DEXi zgolj enega (FastSpring) uvrstil v razred najustrežnejših. Kar štiri kandidate (PayPal, Stripe, BlueSnap in PayLane) je uvrstil v srednji razred, enega (2checkout) pa je označil kot popolnoma neustreznega, kar smo tudi pričakovali. Z rezultatom smo bili zadovoljni, zato smo zmagovalca uporabili za integracijo obeh sistemov. Izbrani kandidat je z vidika integracije najprimernejši, poleg tega pa se s ponujenimi funkcionalnostmi odreže podobno, kot večina konkurentov. Slabost, ki jo moramo vzeti v zakup, so precej visoki stroški posameznih transakcij.



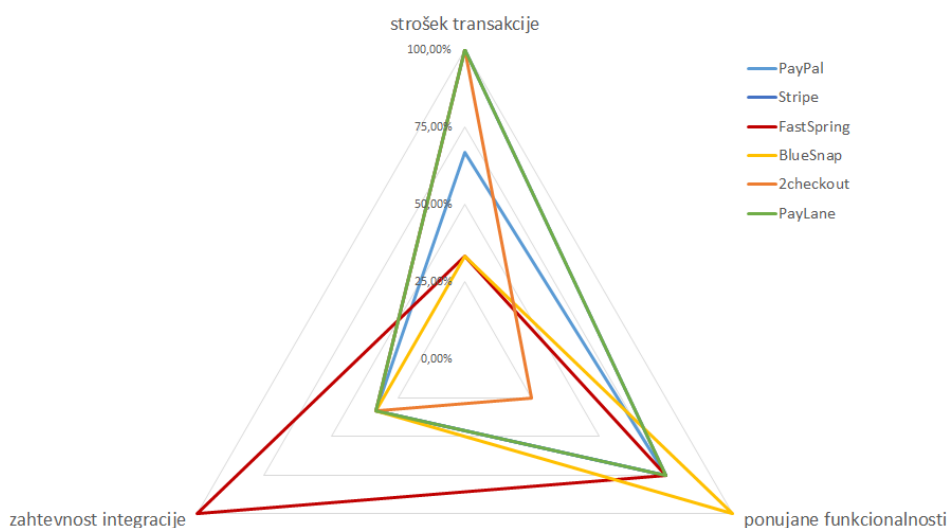
Slika 3.1: Končne ocene kandidatov

	PayPal	Stripe	FastSpring	BlueSnap	2checkout	PayLane
<b>izbira</b>	<b>srednje</b>	<b>srednje</b>	<b>najustrežnejši</b>	<b>srednje</b>	<b>neustrezen</b>	<b>srednje</b>
<b>plačilnega sistema</b>	<b>ustrezen</b>	<b>ustrezen</b>		<b>ustrezen</b>		<b>ustrezen</b>
<i>. strošek transakcije</i>	srednji	nizek	visok	visok	nizek	nizek
<i>. ponujane funkcionalnosti</i>	ena	ena	ena	vse	nobena	ena
<i>. . preverjanje davčne številke</i>	zunanja storitev	zunanja storitev	da	da	ne	ne
<i>. . konfiguracija produktov</i>	nezahtevana	nezahtevana	zahtevana	nezahtevana	zahtevana	nezahtevana
<i>. zahtevnost integracije</i>	visoka	visoka	nizka	visoka	visoka	visoka

Tabela 3.4: Rezultati vrednotenja kandidatov

### 3.6 Analiza kandidatov

Vsakega izmed kandidatov bomo v nadaljevanju na kratko opisali. Rezultate smo predstavili s polarnim grafikonom, na katerem vsako oglišče predstavlja enega izmed kriterijev. Črta bližje robu kaže na boljšo oceno danega kriterija. Površina orisanega lika je sorazmerna s kvaliteto oziroma z ustreznostjo plačilnega sistema. Na grafu je možno videti le pet kandidatov, saj imata kandidata Stripe in PayLane enake vrednosti pri vseh treh atributih.



Slika 3.2: Graf ocene kriterijev za posamezne kandidate

**PayPal** lahko po novem uporabljamo tudi, če pri njih nimamo odprtega računa, kot to omogočajo vsi ostali ponudniki plačilnih storitev. Stranka lahko tako v ustrezno polje vnese številko kartice, ne da bi za to morala najprej opraviti registracijo na PayPalu. To je bil razlog, da smo ga vključili med kandidate. V nadaljnji raziskavi smo ugotovili, da ne ponuja storitve preverjanja davčne številke, a lahko dokaj enostavno uporabimo katero izmed zunanjih storitev, ki to ponujajo. V tabeli 3.5 smo pripravili tudi primerjavo namenskih storitev za preverjanje davčnih števil. Dobra stran PayPala je, da produktov ni potrebno konfigurirati vnaprej. Tudi strošek transakcije spada v srednji razred, vendar pa bi bila časovna zahtevnost integracije velika.



**Stripe** je naslednji kandidat, ki je na trgu dobro uveljavljen. Glede na izbrane kriterije je zelo podoben PayPalu, le da ponuja nižji strošek transakcije. Žal smo kasneje ugotovili, da ga v Sloveniji ni moč uporabljati. V oddelku za podporo so nam zagotovili, da bodo svoje storitve kmalu ponujali tudi pri nas, vendar jih ni bilo smotrno čakati.

**FastSpring** je plačilni servis z najvišjo provizijo, a smo se na koncu odločili zanj. Ponuja izjemno pregleden uporabniški vmesnik, njihova storitev preverjanja davčne številke je najbolj temeljita. Produkte smo morali konfigurirati tudi v njihovi nadzorni plošči, vendar to po drugi strani pomeni veliko lažjo pripravo akcij in kuponov za popuste. Na tem mestu velja pohvaliti tudi njihovo ažurno podporo in trud, ki so ga izkazali pri integraciji z našo platformo za izdajo licenc. Naj še omenimo, da se obrazec za vnos podatkov o plačilu nahaja na njihovi strani, zato moramo stranke preusmerjati. Kljub temu je obrazec popolnoma fleksibilen.

**BlueSnap** je na prvi pogled najprimernejši kandidat. Zaračunava nižjo provizijo od FastSpringa, prav tako ima integrirano storitev preverjanja davčne številke. Produktov ni potrebno konfigurirati v njihovi nadzorni plošči, kar pomeni, da bi žetone po zgoraj opisanem postopku lahko pridobivali programsko. To bi pomenilo precej več dela z integracijo obeh sistemov na začetku, vendar bi bila dolgoročno gledano to najboljša rešitev. V podjetju smo se odločili za hitrejši zagon spletne trgovine, zato smo BlueSnap upoštevali kot najboljšo alternativo. V postopku integracije smo skrbeli za manjše programske module in prilagodljivost. Pustili smo namreč odprto opcijo, da v prihodnosti zamenjamo plačilni servis, če se bo to izkazalo kot nujno potrebno.

**2checkout** ima sicer najnižjo provizijo, a nima integrirane storitve za preverjanje davčne številke. Prav tako zahteva konfiguracijo produktov v njihovi nadzorni plošči. Za njim stoji manjša ekipa in ni bil obravnavan kot resen konkurent. Podobno velja tudi za **PayLane**, čeprav ta ne zahteva konfiguracije produktov v nadzorni plošči.

### 3.7 Ponudniki preverjanja davčne številke

Glede na to, da kar nekaj kandidatov ne ponuja storitve za preverjanje davčne številke, vendar prva in hkrati pomembnejša dva omogočata integracijo zunanje storitve, smo raziskali tudi to področje. Zopet smo poiskali nekaj najpopularnejših ponudnikov in jih primerjali v tabeli 3.5. Postopek je v splošnem tak, da se pri ponudniku preverjanja davčne številke registriramo kot trgovec in ob tem prejmemo svoj identifikacijski ključ. V spletni trgovini dodamo polje za vnos davčne številke. Naslednji korak je lahko v vsaki spletni trgovini izpeljan drugače. Najosnovnejša rešitev je, da stranka ob izpolnitvi vseh polj za spletno plačilo podatke potrdi z gumbom „Pošlji”. V tem trenutku se izvede klic na REST API ponudnika storitve preverjanja davčne številke z vnešeno davčno številko in našim identifikacijskim ključem. Prvi podatek služi kot ključ za iskanje, drugi pa za identifikacijo trgovca, kateremu se v tem trenutku zaračuna provizija. REST API vrne paket podatkov v vnaprej definirani obliki. Po navadi so to podatki o veljavnosti davčne številke, sedež podjetja ter davčna stopnja, ki jo moramo obračunati. Zunanji sistem mora opraviti kar nekaj nalog. Ugotoviti mora, od kod stranka prihaja in na podlagi zakonskih določil določiti davčno stopnjo. Tu veljajo različna pravila glede na sedež stranke in prodajalca. Obravnavati moramo štiri različne scenarije, ki jih zgolj omenjamo, s podrobnostmi pa se v okviru tega dela ne bomo ukvarjali:

- stranka in prodajalec se nahajata v isti evropski državi;
- stranka in prodajalec se nahajata v različnih evropskih državah;
- stranka se nahaja v evropski državi, podjetje pa ne;
- stranka se ne nahaja v evropski državi, podjetje pa.

Podatke, ki nam jih je vrnil REST API, uporabimo v naslednjih korakih izvedbe plačila. Omenjeno rešitev lahko za voljo boljše uporabniške izkušnje izboljšamo. Že ob izgubi fokusa na polju za vnos davčne številke lahko izvedemo klic na REST API. Ob prejemu odgovora lahko stranki z dinamičnim poljem takoj sporočimo, če je vnešena davčna številka veljavna in po kakšni stopnji se jim bo obračunal davek na dodano vrednost.

Kot že omenjeno, smo se na koncu odločili za plačilni servis FastSpring, ki že ponuja storitev preverjanja davčne številke in se nam ni bilo potrebno ukvarjati s tem postopkom.

ponudnik	strošek transakcije	podpora in integracija	vrača podatke o podjetju
Taxamo	2 % ali 0,20 €	Stripe, PayPal, Braintree	DA
Octobat	0,30 € glede na poslovanje	Stripe	DA
TaxCloud	brezplačno	Stripe, PayPal, Google Wallet, Braintree podpira le ZDA	NE
Avalara	ni stroškov	razčlenjevanje spletne strani	NE

Tabela 3.5: Primerjava ponudnikov preverjanja davčne številke

Taxamo <sup>1</sup> je prvi ponudnik, ki omogoča avtomatsko integracijo s tremi večjimi plačilnimi servisi. To pomeni, da v izbranem plačilnem servisu le vnesemo svoje uporabniške podatke, ki smo jih prejeli pri Taxamu in s tem pridobimo podobno funkcionalnost, kot jo na primer sam po sebi nudi FastSpring. Ni nam potrebno programirati klicev na REST API, ker za to poskrbita Taxamo in plačilni servis. Vrača nam tudi podatke o podjetju (naziv in sedež), zato lahko ta polja avtomatsko izpolnemo in olajšamo delo stranki.

Octobat <sup>2</sup> je naslednji ponudnik, ki se od prvega ne razlikuje veliko, le da ga je mogoče integrirati zgolj v plačilni servis Stripe. Tudi cenovno sta si zelo podobna, saj Octobat zaračunava provizijo glede na obseg poslovanja.

TaxCloud <sup>3</sup> je ponudnik, ki storitev preverjanja davčne številke ponuja brezplačno. Žal pozna le davčne številke iz Amerike, zato za nas ne pride v poštev.

Avalara <sup>4</sup> omogoča izračun davčne stopnje na njihovi spletni strani, REST API-ja pa ne ponujajo. To pomeni, da bi morali razčlenjevati (angl. parsing) njihovo spletno stran, da bi prišli do podatkov. To se v praksi prav gotovo ne izkaže za dobro rešitev. V primeru najmanjše spremembe v strukturi rezultatov bi naše razčlenjevanje odpovedalo. Sistem za prodajo licenc bi prenehal delovati zaradi izračuna davčne stopnje. Tega si nobeno resno podjetje ne more in ne sme privoščiti, zato se za kaj takega ne bi odločili.

---

<sup>1</sup><https://www.taxamo.com/>

<sup>2</sup><https://www.octobat.com/>

<sup>3</sup><https://taxcloud.net/>

<sup>4</sup><https://www.avalara.com/products/sales-and-use-tax/avatax-2/>

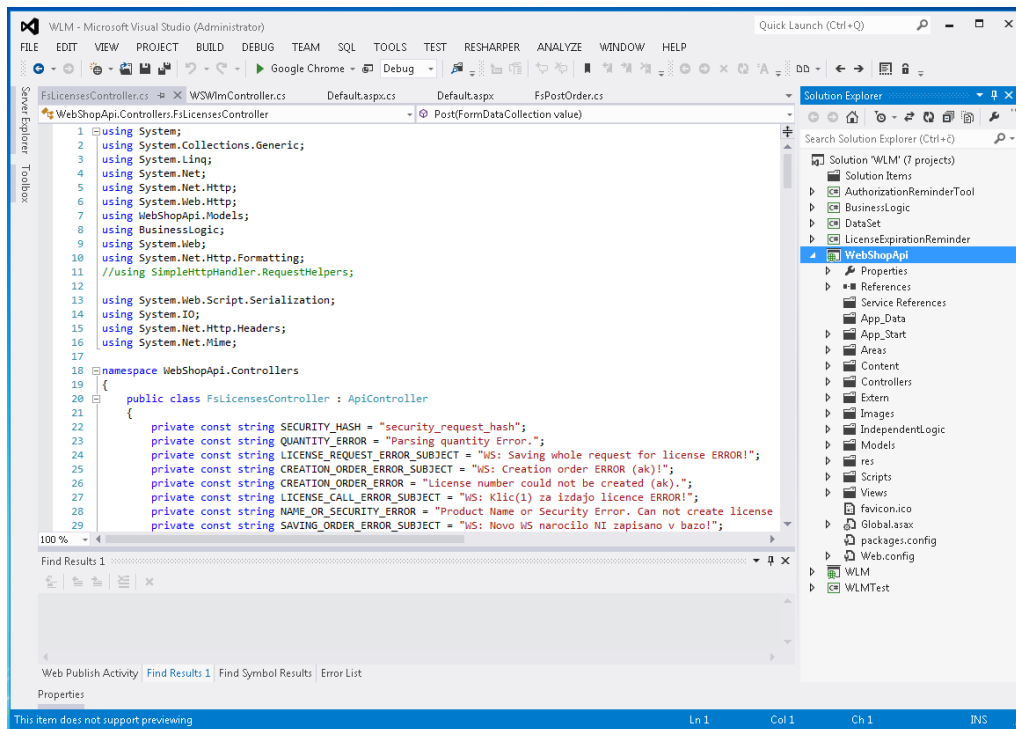
## Poglavje 4

# Pregled tehnologij in orodij

Integracijo plačilnega sistema s portalom za izdajo licenc smo izvedli s pomočjo določenih orodij in v izbranem programskem jeziku. Preden se lotimo obravnave celotnega procesa, si na kratko oglejmo uporabljene tehnologije.

### 4.1 Microsoft Visual Studio

Microsoft Visual Studio je Microsoftovo integrirano razvojno okolje (IDE). Uporablja se za razvoj računalniških programov za Microsoft Windows, kot tudi spletnih strani, spletnih aplikacij in spletnih storitev. Urejevalnik kode vključuje podporo za samodejno dokončanje ukazov (angl. *intellisense*). Vgrajen ima razhroščevalnik (angl. *debugger*). Ima še vrsto ostalih vgrajenih orodij, npr. za gradnjo grafičnih uporabniških vmesnikov, shem podatkovnih baz itd. Z ustreznim vtičnikom pridobimo možnost verzioniranja izvorne kode. Podpira različne programske jezike: C, C++, VB.NET in C#. Podpora drugih jezikov, kot so Python, Ruby, Node.js, je na voljo preko ločeno nameščenih programskih paketov. Prav tako podpira XML / XSLT, HTML / XHTML, JavaScript in CSS.



Slika 4.1: Visual Studio

## 4.2 Programski jezik C#

Programski jezik C# je ustvarjen z namenom, da bi bil preprost, varen, moderen, objektno orientiran, visoko kvaliteten in interno usmerjen programski jezik.

V marsičem je podoben programskima jezikoma Java in C++. Je zmogljiv kot jezika C in C++ in urejen ter objekten kot Java ali npr. Pascal (Delphi) [1].

### 4.3 ASP.NET MVC - WebApi

ASP.NET MVC je platforma za razvoj spletnih aplikacij, ki temelji na ASP.NET in implementira vzorec Model-View-Controller (model-pogled-krmilnik). Model predstavlja stanje določenega vidika aplikacije. Kontroler obdeluje interakcije z uporabnikom in ustrezno posodablja model, da ustreza spremembi stanja aplikacije, nato pa posreduje podatke pogledu. Pogled od kontrolerja prejme potrebne informacije in zgradi uporabniški vmesnik ter ga prikaže uporabniku. Ločitev razvoja na omenjene tri vidike omogoča večjo modularnost in fleksibilnost, hitrejši razvoj aplikacij ter njihovo lažje vzdrževanje [23].

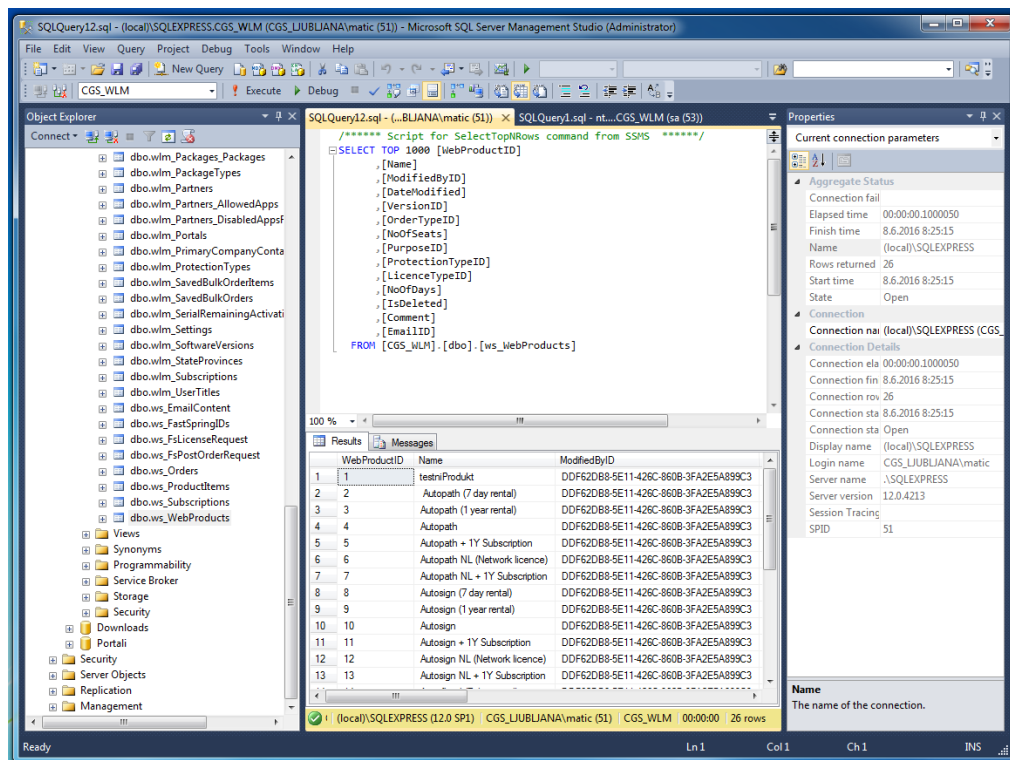
Protokol HTTP ne služi zgolj serviranju spletnih strani, temveč je tudi zmogljiva platforma za gradnjo API-jev, ki izpostavljajo storitve in podatke. Je preprost, prilagodljiv in vseprisoten. Skoraj vse platforme imajo knjižnico za podporo omenjenemu protokolu, zato lahko storitve HTTP dosežejo širok spekter odjemalcev, vključno z brskalniki, mobilnimi napravami in tradicionalnimi namiznimi aplikacijami. ASP.NET Web API je ogrodje (angl. framework) za gradnjo spletnih vmesnikov API na podlagi ogrodja .NET, ki ga bomo na kratko predstavili v naslednjem odstavku [8]. Njegov cilj je doseganje čim večjega števila odjemalcev in ob tem slediti konceptu spletnih storitev REST. Ima dobro podporo za vračanje ustreznih statusnih kod HTTP (405, 201 ...). Že samo ogrodje podpira serializacijo sporočil XML in JSON v objekte. Na enak način kot kontrolerji MVC omogoča anotacije metod in razredov, s katerimi zagotovimo posebno obnašanje ([Authorize], [RequireHttps] idr.).

Ogrodje .NET Framework je Microsoftov programirni model upravljanja programske kode (angl. managed code) in računalniško okolje (angl. platform) za razvoj aplikacij na Microsoftovih odjemalcih, strežnikih in mobilnih ter vgradnih napravah. Ogrodje ponudja pregledno, konsistentno objektno orientirano programiranje s poudarkom na varnosti in visoki produktivnosti [1].

## 4.4 Microsoft SQL Server

Strežnik Microsoft SQL je sistem za upravljanje relacijskih zbirk podatkov. Razvil ga je Microsoft, zato se dobro ujema z njihovimi ostalimi strežniškimi proizvodi. Njegov primarni programski jezik je Transact-SQL, ki je implementacija standarda ANSI/ISO SQL. Uporabljajo ga tako pri Microsoftu kot tudi pri Sybase [24]. Za potrebe diplomske naloge smo uporabili Express različico, ki je na voljo za brezplačno komercialno rabo.

SQL Server Management Studio je grafični uporabniški vmesnik za upravljanje in administracijo SQL Server podatkovnih baz. Z njim lahko ustvarjamo in brišemo podatkovne baze in tabele ter načrtujemo baze z risanjem ER diagramov. Enostavno lahko tudi vnašamo, urejamo in brišemo podatke v posameznih tabelah. Omogoča ustvarjanje varnostnih kopij podatkovnih baz na enostaven način.



Slika 4.2: SQL Server Management Studio



## Poglavje 5

# Platforma za izdajo licenc

Podjetje uporablja spletno aplikacijo (v nadaljevanju partnerski portal), ki v osnovi služi za upravljanje z naročili proizvedene programske opreme. V njej lahko oddamo novo naročilo, pregledujemo podatke o preteklih naročilih, prenesemo namestitvene datoteke izbrane programske opreme. Prav tako lahko upravljamo s partnerji podjetja in z uporabniki, ki imajo dostop do spletne aplikacije. Kreiramo in upravljamo lahko programske module in pakete. Upravljamo lahko z verzijami programske opreme in vnašamo prevode le-te v jezike, ki jih podpiramo. Aplikacija ponuja mnogo funkcionalnosti. Osredotočili se bomo zgolj na tiste, ki jih dejansko potrebujemo za namen avtomatizacije izdaje licenc.

### 5.1 Določitev prodajnega partnerja

V partnerskem portalu se nahajajo vsi partnerji, ki za nas prodajajo programsko opremo. Tako lahko sami kreirajo novo naročilo in pridobijo licenco, ki jo posredujejo stranki. Portal izda račun partnerju, hkrati pa o novem naročilu obvesti računovodstvo. Za potrebe spletne trgovine oziroma avtomatske izdaje licenc smo dodali novega partnerja z imenom „CGSWeb-ShopDealer“. Gre za navideznega partnerja, ki smo ga dodali zgolj zaradi boljšega pregleda nad naročili. Podjetje svojo ponudbo predstavlja na ločenih

spletnih straneh za slovenski, mednarodni in ameriški trg. Spletno prodajo smo zaenkrat omogočili na zadnjih dveh. Vsako naročilo v partnerskem portalu mora biti povezano z enim od uporabnikov. Iz analitskih razlogov želimo ločevati med naročili z mednarodne in ameriške spletne strani. V ta namen smo dodali dva nova, zopet navidezna uporabnika, z imenoma „WebShopUserINT” in „WebShopUserUSA”.

## 5.2 Konfiguracija produktov za prodajo


V naslednjem koraku je potrebno konfigurirati produkte, ki se bodo prodajali v spletni trgovini. Ti produkti morajo biti v celoti prednastavljeni. Ko bo plačilni servis dostopal do našega REST API-ja in zahteval licenco za izbrani produkt, mora partnerski portal poznati nastavitve za ta produkt in glede na to izdati ustrezno licenco. Nastavitve vključujejo tip licence, število aktivacij, vključene pakete itd. V ta namen smo pripravili nov vmesnik, ki bo omogočal natančno konfiguracijo produktov in jih zapisal v podatkovno bazo za kasnejšo uporabo. Za boljše razumevanje prilagamo sliko 5.1, ki je posnetek zaslonske maske, v kateri kreiramo ali urejamo produkt, namenjen prodaji v spletni trgovini. Določiti mu moramo smiselno ime, da se med produkti lažje znajdemo. Nastavimo, ali gre za nakup nove licence, za nakup nove licence z naročnino na posodobitve, ali za najem licence za določeno obdobje. Vsaki licenci določimo, koliko aktivacij ji dovolimo. Ponujamo tudi mrežne licene, ki imajo drugačen postopek aktivacije, zato moramo podatek o tipu zapisati v bazo.

Sledi skupina nastavitvev, ki zadevajo vključevanje paketov v končni produkt. Naša programska oprema je precej modularna, zato lahko v sklopu enega produkta ponudimo več paketov. Izdamo lahko stalno licenco, ali pa licenco, ki po določenem času poteče. Če smo izbrali nakup licence z naročnino na posodobitve, lahko posebej nastavimo še, koliko časa traja naročnina. Po izteku tega obdobja je lahko licenca za produkt še vedno veljavna, le ukinjena je možnost brezplačnega posodabljanja programske opreme na novejšo

verzije.

Pojasniti je potrebno še nastavitve, ki vplivajo na komunikacijo s plačilnim servisom. Produktu moramo določiti enolični „FastSpring ID”. To je identifikacijski ključ, ki ga plačilni servis (v nadaljevanju FastSpring) pošlje ob klicu na REST API, ko želi izdajo nove licence. S tem ključem pove, za kateri produkt želi licenco. Takrat v bazi poiščemo skonfiguriran produkt z omenjenim ID-jem in zanj izdamo licenco. Izpolniti je potrebno tudi polje FastSpring Private Key. Gre za skriti ključ, ki ga poznata le FastSpring in naša platforma za izdajo licenc. Vsakemu produktu pripada enolični skriti ključ. Ob klicu FastSpringa na naš REST API z ustrezno transformacijo tega ključa avtoriziramo plačilni servis. Podrobneje bomo o postopku govorili v naslednjem poglavju. Tu smo le pojasnili, za kakšen podatek gre. Ostalo nam je še polje, v katerem izberemo, katero prednastavljeno besedilo bomo poslali kupcu na elektronski naslov, ko bo opravil nakup. Tudi o tem bomo podrobneje govorili v nadaljevanju.

Vsakemu produktu lahko dodamo še komentar za interno uporabo, ki olajša razumevanje konfiguriranih produktov.

 CGS Partner Portal Matic Novak

Home > Admin CGS Web Shop > View CGS Web Shop Products > Create/Edit Web Product

### Create/Edit Web Product

Web Shop product details

Web Product Name: Autopath + Autosign

New/Upgrade: New license with subscription

No. of seats: 1

Protection type: Software - Workstation

Email Order Confirmation Text: Autopath

FastSpring ID:

FastSpring Private Key:

Packages

Items

Autopath

Swept Path Analysis

Autosign

Autosign

Licence type and expiration time in days

Permanent 0

Add new packages:

Autosign

Add

Clear list

Subscription details

Add subscription to the Web Shop Product: ☒

Nr. of days: 365

Comment

Click here to add a comment.

Save Web Product

Slika 5.1: Kreiranje produkta v partnerskem portalu

### 5.3 Prednastavljanje obvestilnih sporočil

Stalna praksa je, da ob nakupu katerega koli izdelka ali storitve preko spleta prejmemo elektronsko sporočilo, v katerem se nam prodajalec zahvali za zaupanje in nas obvesti, kako najbolje uporabljati kupljen element. Tudi mi smo se potrudili, da bi bila poslana sporočila prijazna in uporabna, zato smo jih prilagodili glede na kupljene izdelke. Ustvarili smo bazo vnaprej pripravljenih osnutkov sporočil. V fazi kreiranja produktov v partnerskem portalu izberemo, kateri osnutek sporočila pripada določenemu produktu, kot smo zapisali v zadnjih vrsticah prejšnjega odseka. Implementirali smo dodatno funkcionalnost, ki omogoča pripravo in urejanje omenjenih osnutkov.

Osnutki so sestavljeni iz večih delov. Na začetku se kupcu zahvalimo za izkazano zaupanje in nakup. V naslednjem poglavju izpišemo vse podatke o opravljenem naročilu: številka naročila, datum naročila, serijska številka, število možnih aktivacij izdelka, podatki o stranki, podatki o naročenih produktih, verzija produktov, tip licence (stalna, stalna z naročnino na posodobitve, najem) itd. Pod njimi se nahajajo povezave na prenos kupljenih izdelkov in povezava na navodila za namestitev in aktivacijo.

Trudimo se za čim boljšo razpoznavnost naše programske opreme. Vodilo nam je enostavnost uporabe, zato stremimo k dobri uporabniški izkušnji. V ta namen smo posneli kar nekaj primerov uporabe pomembnejših funkcionalnosti, ki so vključene v programske rešitve. Pogosto izvajamo spletne seminarje, v katerih podrobneje predstavljamo produkte. Prav tako imamo za vsak produkt podrobna pisna navodila s posnetki zaslona. Na konec osnutkov elektronskih sporočil pripnemo povezave do vseh posnetkov in navodil, ki zadevajo kupljene produkte. Dodamo še povezavo, na kateri je dosegljiva tehnična podpora podjetja in skupaj s pozdravom sporočilo pošljemo kupcu.

## 5.4 Pregled spletnih nakupov

V partnerski portal smo dodali funkcionalnost, ki omogoča pregled spletnih naročil. Na vrhu imamo polje za iskanje po naročilih. Dodatno imamo možnost filtriranja naročil glede na izbrano časovno obdobje. Na seznamu se prikažejo ključni podatki, kot so:

- čas naročila;
- podjetje, ki je kupilo izbrano programsko opremo;
- uporabnik partnerskega portala, ki je naročilo kreiral (tako ugotovimo, ali je uporabnik izvedel naročilo na mednarodni ali ameriški spletni strani);
- znesek celotnega naročila;
- naročeni paketi;
- zgenerirana licenčna številka.

V zadnjem stolpcu se nahajajo različni gumbi. Prvi služi prenosu certifikata z licenčno številko, ki ga stranke po navadi natisnejo in shranijo v papirnati obliki. Drugi gumb kaže na prenos datoteke o potrditvi naročila, ki jo najpogosteje potrebujejo v računovodstvu. Tretji gumb služi prenosu računa. Četrty in peti služita urejanju oziroma brisanju naročila.

The screenshot displays the 'CGS Partner Portal' interface. At the top, there is a navigation bar with 'Home', 'Orders', 'Reports', and 'Placed orders'. Below this, a 'Placed orders' section includes a filter input field and a list of placed orders. The list is filtered by 'Dealer: web shop dealers', 'Company: all companies', and 'User: all users'. The date range is set from '09.05.2016' to '16.05.2016'. The table below shows the details of the orders, including Date, Dealer, Company, Buy Price, Order ID, Partner order ID, User, and Content. The total buying price is 0.0000 EUR.

Date	Dealer	Company	Buy Price	Order ID	Partner order ID	User	Content
11.05.2016 12:50	CGS plus d.o.o.	CGS plus d.o.o.	0,00 EUR	6665		WebShopUser2INT	570d-4f6e-609a-baf6 Aquaflood
10.05.2016 16:49	CGS plus d.o.o.	A.D. Servo Mihalj Inženjering	0,00 EUR	6660		WebShopUser2INT	38c4-1481-0bb5-06d0 Autosign
10.05.2016 16:48	CGS plus d.o.o.	A.D. Servo Mihalj Inženjering	0,00 EUR	6659		WebShopUser2INT	4e8e-8883-68b8-2905 Autopath
10.05.2016 16:47	CGS plus d.o.o.	A.D. Servo Mihalj Inženjering	0,00 EUR	6658		WebShopUser2INT	8c48-7bd8-01d7-0e26 Aquaflood
09.05.2016 13:37	CGS plus d.o.o.	CGS plus TEST	0,00 EUR	6652		WebShopUser2INT	ee8b-37e3-a7b2-8b41 Autosign

Buying price (total): 0.0000 EUR

Slika 5.2: Seznam spletnih naročil v partnerskem portalu

## 5.5 Pregled zahtevkov z REST API-ja

V naslednjem poglavju bomo podrobno predstavili komunikacijo med spletnim portalom za izdajo licenc in plačilnim servisom. Na tem mestu bi radi omenili le, da si sporočila izmenjujeta preko REST API-ja. Zaradi varnosti in vpogleda v zgodovino komunikacije v primeru kakršnih koli težav, vsa izmenjana sporočila shranjujemo v podatkovno bazo. V primeru napak bo le-te prav gotovo reševala oseba z ustreznim znanjem, zato bi omenjene zahteve lahko poiskala v bazi. Kljub temu iskanje napake močno poenostavimo, če lahko do sporočil dostopamo preko aplikacije na uporabniku prijazen način. Sporočila lahko ustrezno formatiramo in obarvamo, da so podatki lažje berljivi. V ta namen smo v partnerski portal dodali možnost pregledovanja izmenjanih sporočil. Teh se bo v relativno kratkem časovnem obdobju nabralo ogromno, zato smo poskrbeli za odstranjevanje. V nasprotnem primeru bi želeli iz baze naložiti celoten seznam sporočil, kar bi povzročilo sesutje strežnika.





## Poglavje 6

# Avtomatizacija izdajanja licenc

Podrobno smo razčlenili spletne plačilne sisteme, prav tako smo predstavili platformo za izdajo licenc. Prišli smo do ključne točke diplomskega dela - obe storitvi moramo povezati, da bosta sposobni komunicirati med sabo. Po pregledu dokumentacije plačilnega sistema smo ugotovili, da bomo komunikacijo najlažje in najučinkoviteje implementirali po protokolu REST.

### 6.1 Spletne storitve

Spletne storitve so način izmenjave podatkov in informacij preko omrežja. Ključno pri njih je, da so neodvisne od tehnologije in platforme, dostopne pa so preko interneta. Določa jih enolični krajevnik vira (URL). So modularne in dinamične narave. Opisane in objavljene so v javnih imenikih UDDI. Zasnovane so lahko na podlagi dveh arhitekturnih stilov, to sta SOAP ali REST. Pomembno je, da izberemo ustrezno arhitekturo. Spletne storitve objavljajo svoje funkcionalnosti preko WSDL-ja, če izberemo SOAP, oziroma preko URI-ja, če uporabimo REST. Nista omejena na noben poseben protokol. Uporabimo lahko SMTP, FTP, HTTP, in tako dalje, vendar se najpogosteje uporablja protokol HTTP.

REST je arhitekturni stil načrtovanja spletnih aplikacij. Opisal ga je Roy Fielding v svoji doktorski disertaciji leta 2000 [7]. Oblika sporočila je lahko

tako v formatu XML, kot tudi v formatu JSON. XML je precej obsežen, zato vpliva na omrežni promet in na obremenitev strežnika. V analizi je bilo ugotovljeno, da ima REST boljšo prepustnost in odzivni čas od SOAP-a. Gre za še razvijajočo se tehnologijo, ki jo uporablja že ogromno aplikacij [13]. Izjemno pridobiva na popularnosti, zato je vedno večja dilema in pogosto tema razprav, kako oblikovati REST FULL spletno storitev, ki bo omogočala prilagodljivost, razširljivost in interoperabilnost. Težava je v pomanjkanju standarda, zato nastajajo REST API-ji, ki ne ustrezajo smernicam REST-a [14]. Glavne značilnosti REST-a so prilagodljivost, zanesljivost, varnost, hitrost in optimalna cena [26].

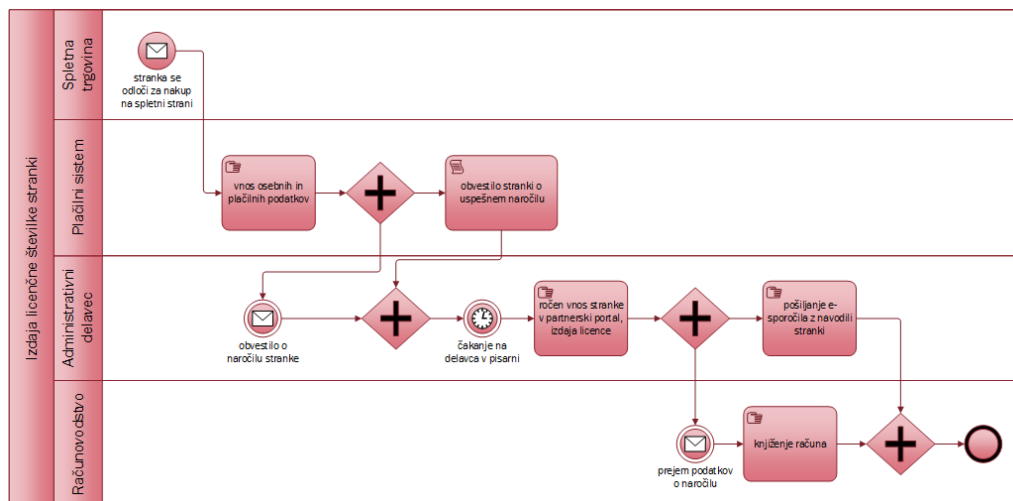
## 6.2 Prenova poslovnega procesa

Z integracijo obeh sistemov bomo prenovili poslovni proces v podjetju, zato si ga podrobneje pogledjmo. Spremembe so prikazane na sliki 6.2.

Najprej pogledjmo obstoječe stanje na sliki 6.1. Stranka lahko na spletni strani podjetja pregleduje ponudbo programske opreme. V primeru, da se odloči za nakup, klikne na gumb „kupi”, ki se nahaja poleg izbranega produkta. Stran jo preusmeri na spletni plačilni sistem PayPal, kjer vnese osebne podatke ter podatke za plačilo. Če je transakcija uspešno opravljena, stranka o tem prejme obvestilo. V njem piše, naj počaka na izdajo licence, za katero bo naslednji delovni dan poskrbelo osebje podjetja. O uspešni transakciji je obveščeno tudi podjetje, saj mora nekdo od zaposlenih naročilo obdelati. Obdelava naročila pomeni ročen vnos podatkov o stranki in produktu v partnerski portal, ko je eden od administratorjev na voljo. Partnerski portal administratorju vrne licenčno številko. Ta jo skupaj z navodili za namestitve in uporabo kupljene programske opreme posreduje stranki po elektronski pošti. O novem naročilu obvesti še računovodstvo, da poknjiži transakcijo.

Moteče in zamudno je že ročno vnašanje podatkov o stranki in o kupljenih produktih v partnerski portal. Zaradi tega se pojavi tudi večja možnost

vnosnih napak. Še toliko bolj nedopustno pa je čakanje stranke, da prejme licenčno številko od administratorja, ko bo ta našel čas za obdelavo naročila.



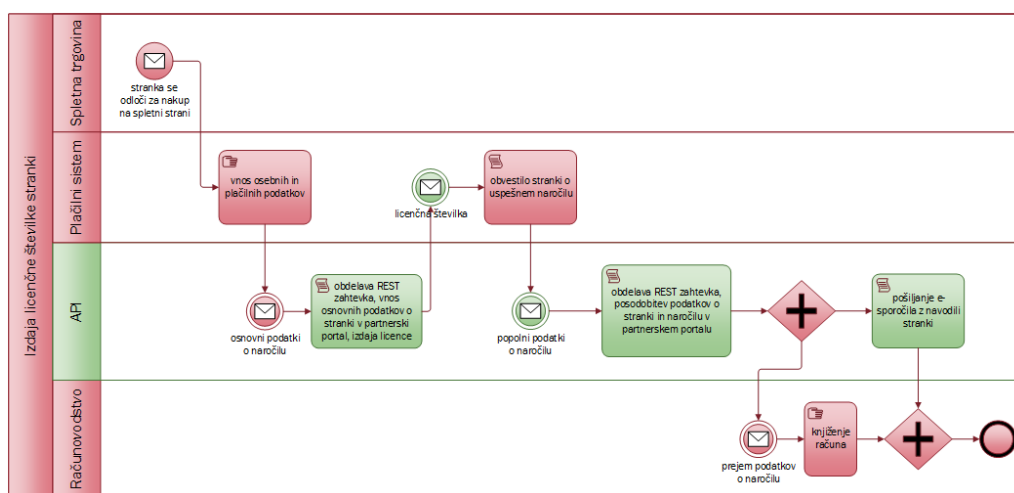
Slika 6.1: Diagram prvotnega poslovnega procesa

Glede na to, da stremimo k bogati uporabniški izkušnji, bi bilo pričakovano, če bi stranka lahko tudi na praznik oziroma nedelovni dan preko spleta kupila izbrano programsko opremo, licenčno številko pa bi v nekaj sekundah prejela v svoj elektronski predal. S tem bi se stranka izognila čakalni dobi, administrator pa bi lahko svoj čas namenil ostalim zadolžitvam, saj mu ne bi bilo potrebno vnašati podatkov v partnerski portal. S prenovo poslovnega procesa bomo poskušali doseči ravnokar opisano stanje.

Prvi korak scenarija bo po novem skoraj identičen staremu. Stranka bo tudi v tem primeru ob kliku na gumb za nakup preusmerjena na spletni plačilni sistem, tokrat FastSpring. Vnesla bo osebne podatke in podatke za plačilo. V primeru, da bo prenos denarja s strankinega na trgovčev račun uspešen, bo plačilni servis FastSpring na API podjetja poslal zahtevo za izdajo licence. Potek naročila si bomo podrobneje ogledali v nadaljevanju, zaenkrat naj predstavimo zgolj koncept. API bo poskrbel za zapis stranke v partnerski portal in FastSpringu vrnil zgenerirano licenco. Stranka bo prejela obvestilo, da je bilo naročilo uspešno zaključeno. V tem trenutku

bo FastSpring ponovno sprožil klic na naš API, tokrat z vsemi podatki o naročilu. API bo poskrbel za posodobitev podatkov o stranki in naročilu v partnerskem portalu in stranki poslal elektronsko sporočilo. V njem bosta licenca ter navodila za namestitev in uporabo kupljene programske opreme. Partnerski portal bo računovodstvo avtomatsko obvestil o novem naročilu, da bo pooblaščen oseba poknjžila transakcijo.

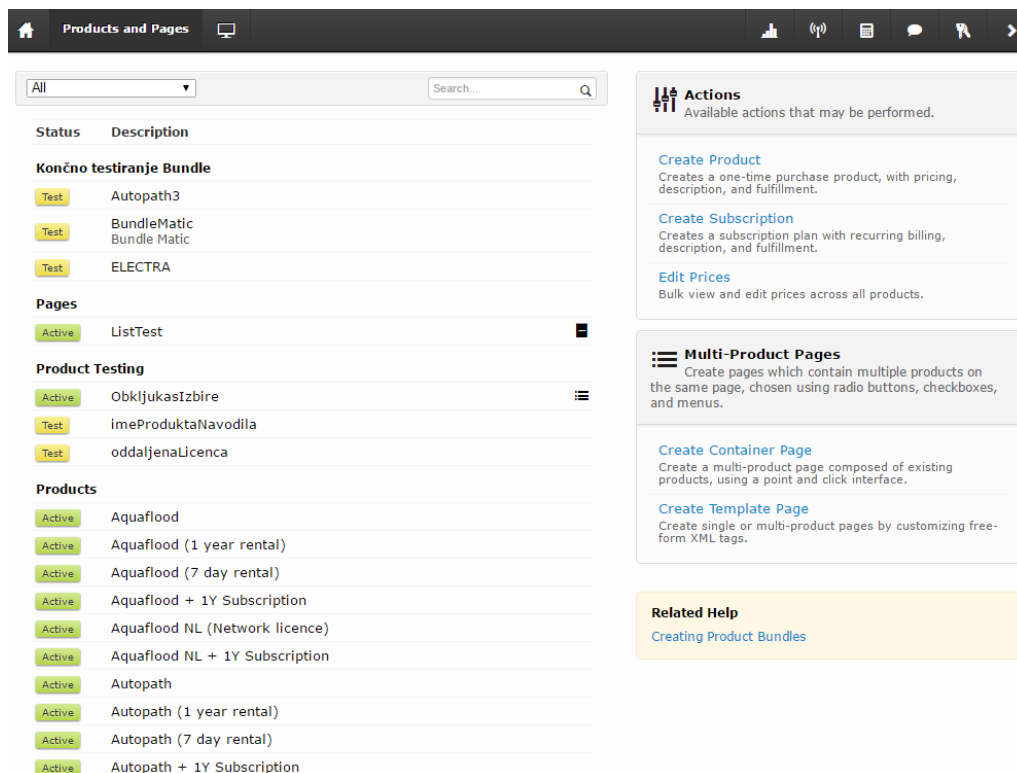
Tako odstranimo nepotrebno čakanje in ročno vnašanje podatkov.



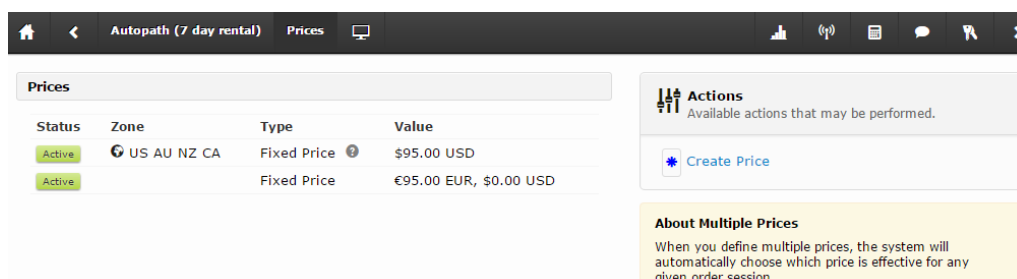
Slika 6.2: Diagram novega poslovnega procesa

## 6.3 Identifikacija produktov

Plačilni servis se mora zavedati, kateri produkt bo stranki prodal. Glavni razlog je ta, da mora od platforme za izdajo licenc pridobiti ustrezno licenčno številko, ki bo aktivirala izbrani produkt. V ta namen smo na nadzorni plošči plačilnega servisa kreirali produkte, podobno, kot smo to storili v partnerskem portalu. Tukaj se nismo ukvarjali s tipom licenc, časom veljavnosti in ostalimi nastavitvami, ker vse to pozna partnerski portal. Nastavili smo imena produktov, njihov kratek opis in logotip. Dodali smo njihove cene in naslov našega API-ja, kamor je ob naročilu potrebno poslati klic za izdajo licence. Na tem mestu velja omeniti dodatne funkcionalnosti, ki jih ponuja izbrani plačilni servis za določanje cen produktov. Vsaki ceni lahko določimo več pogojev. Omenili smo, da za prodajo uporabljamo dve ločeni spletni strani, ki pokrivata mednarodni in ameriški trg, zato bi lahko cene določali glede na to, s katere spletne strani je bila stranka preusmerjena na plačilni servis. Poleg tega FastSpring omogoča tudi nastavljanje pogoja ceni, ki na podlagi IP naslova preveri, iz katere države stranka dejansko prihaja. Ne glede na to, s katere spletne strani je bila stranka preusmerjena, bo zanjo cena drugačna, če prihaja iz Amerike, kot, če prihaja npr. iz Indije. Funkcionalnost smo s pridom izkoristili, saj cene produktov prilagajamo trgu. Vrnimo se še na API, ki skrbi za izdajo licenc. V nastavitvah vsakega produkta smo dodali naslov tega API-ja. Ob shranjevanju nastavitvev nam FastSpring vrne dvaintrideset mestni zasebni varnostni ključ. Tega poznata le FastSpring in naš partnerski portal. Z njim poskrbimo, da lahko za izdajo licenc zaprosi zgolj FastSpring in nihče drug. Ime produkta, ki služi kot identifikacijski ključ in ravnokar omenjeni zasebni ključ, moramo zato dodati še k produktu v partnerskem portalu. Tako sta produkta v obeh sistemih med sabo povezana. Kako poteka komunikacija med obema storitvama in kje točno uporabimo omenjeni varnostni ključ, si bomo podrobno ogledali v naslednjem poglavju.



Slika 6.3: Nadzorna plošča plačilnega servisa za urejanje in dodajanje produktov



Slika 6.4: Določanje cen in pogojev posameznim produktom

**General** | Security | Parameters | Advanced

**URL**  
  
Remote URL the system will access to generate a license. It must begin with http:// or https://.

**Method**  
   
HTTP method used to access the remote URL.

**POST Encoding**  
   
Character encoding to use for the POST method. Not applicable to GET.

**Output Format**  
   
The type of license format the system should expect from the remote server.

**Actions**  
Available actions that may be performed.

- [Save](#)
- [Run Test](#)

**Related Help**  
[MIME 1.0 License Output Format](#)

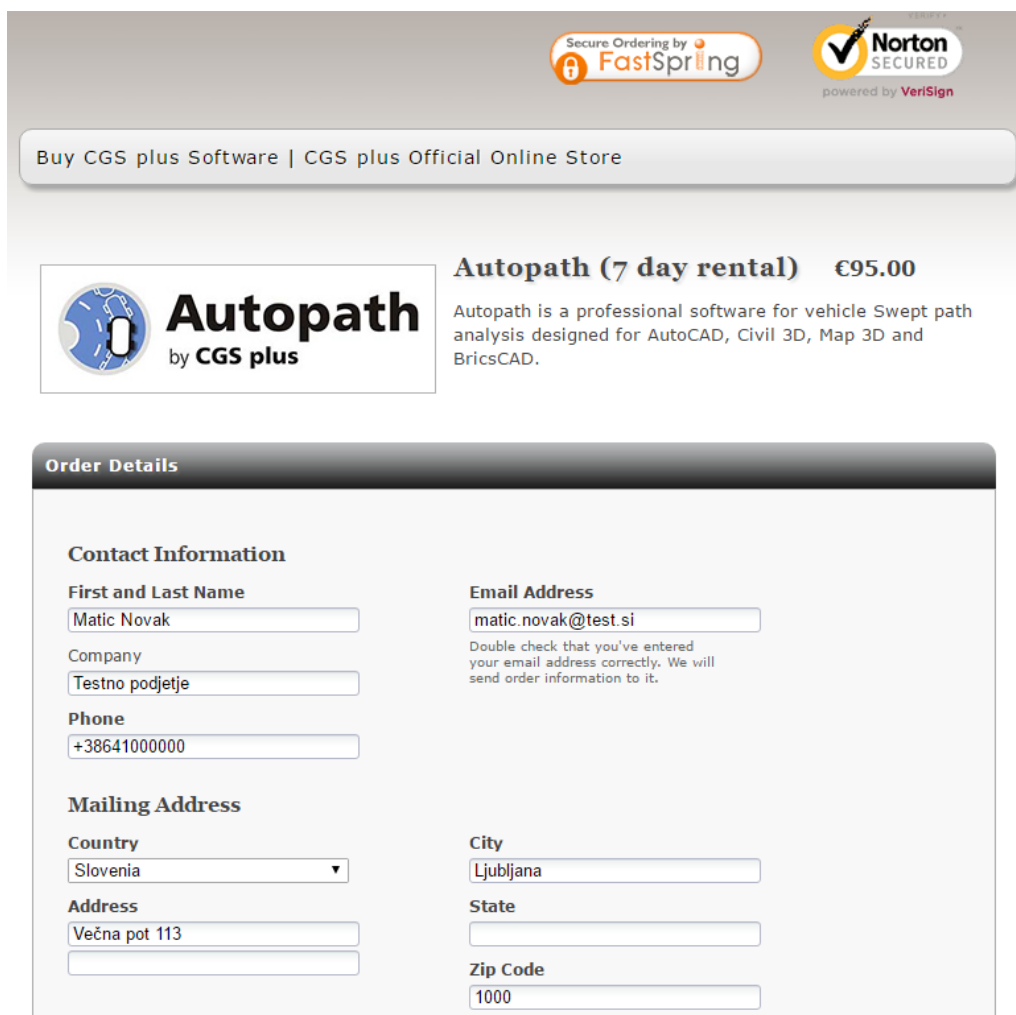
Slika 6.5: Nastavitve za oddaljeno generiranje serijskih števil

## 6.4 Potek naročila

### 6.4.1 Izbira produktov in vnos podatkov


Naročilo se prične, ko stranka na spletni strani podjetja bodisi mednarodni bodisi ameriški, klikne na gumb „kupi“. Takrat je preusmerjena na prodajno stran produkta, ki gostuje na strežniku plačilnega sistema. V glavi strani se nahajata logotip in kratek opis produkta. Sledi cena, ki je določena glede na kriterij, omenjen v prejšnjem poglavju. Pod njo se nahaja obrazec za vnos osebnih podatkov o stranki (glej sliko 6.6) in obrazec za izbiro ter vnos plačilnih podatkov (glej sliko 6.7). Stranka se z gumbom „naprej“ premakne v predzadnji korak košarice (glej sliko 6.8), kjer se ji izpiše izbrani produkt in končna cena. Tu se zopet nahajata dve vnosni polji. Prvo je namenjeno davčni številki. V primeru, da je kupec pravna oseba, vnese svojo davčno številko in pritisne na gumb za osvežitev košarice. Glede na državo in ostale pogoje, se stranki obračuna ustrezna stopnja DDV-ja, ki se prišteje skupnemu znesku košarice. Drugo vnosno polje služi za vnos promocijske kode. Podjetje se lahko odloči za pripravo akcij ob posebnih dogodkih in v nadzorni plošči FastSpringa ustvari seznam promocijskih kod, ki jih razdeli med stranke. V kolikor stranka ima kodo, jo vpiše v omenjeno vnosno polje in ponovno

pritisne na gumb za osvežitev košarice. Obračunan popust se ji odšteje od skupnega zneska košarice. Posredovanje stranke se konča s klikom na gumb za zaključek naročila.



Secure Ordering by **FastSpring** **Norton SECURED** powered by VeriSign

Buy CGS plus Software | CGS plus Official Online Store

**Autopath**  
by CGS plus

**Autopath (7 day rental) €95.00**

Autopath is a professional software for vehicle Swept path analysis designed for AutoCAD, Civil 3D, Map 3D and BricsCAD.

### Order Details

**Contact Information**

**First and Last Name**

**Company**

**Phone**

**Email Address**

Double check that you've entered your email address correctly. We will send order information to it.

**Mailing Address**

**Country**

**City**

**Address**

**State**

**Zip Code**





Slika 6.6: Naročilo - vnos osebnih podatkov



**Payment Method**

Test mode activated for this session only. To reset mode, quit / close **all** browser windows.

☒ Test Payment

☐     Credit Card

☐ Wire Transfer

☐ Check or Money Order

☐ Purchase Order

**Next** ➔



Slika 6.7: Izbira plačilne metode in vnos plačilnih podatkov


Secure Ordering by **FastSpring**


**Norton SECURED**  
powered by VeriSign

Buy CGS plus Software | CGS plus Official Online Store

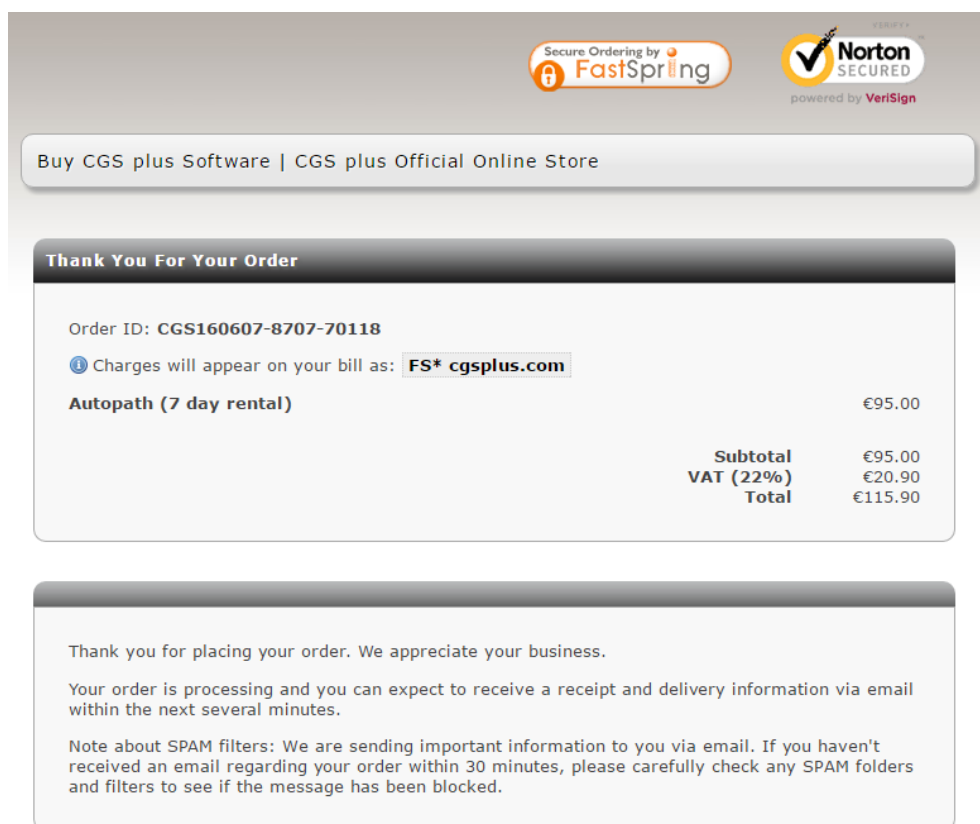
**Your Order**

<b>Autopath (7 day rental)</b>	€95.00
<b>Subtotal</b>	€95.00
<b>VAT (22%)</b>	€20.90
 <a href="#">Update VAT ID</a> 	<input type="text"/>
<b>Total</b>	€115.90

 [Update Coupon](#)

 **Complete Order**

Slika 6.8: Vnos davčne številke in promocijske kode



Slika 6.9: Obvestilo o zaključenem naročilu

### 6.4.2 Generiranje licenčne številke

V tem koraku se izvrši poslovna logika, ki je bila ključni predmet diplomske naloge. FastSpring sproži REST (HTTP POST) klic na naš API z nekaj osnovnimi informacijami o naročilu v obliki slovarja s ključem in vrednostjo: elektronski naslov in naziv stranke, naziv izbranega produkta, referenčna številka naročila in dvaintrideset mestni varnostni izveček. Poslovna logika API-ja s pomočjo naziva izbranega produkta v bazi najprej poišče produkt in njegove nastavitve, za katerega je potrebno izdati licenco. V primeru, da izbrani produkt ne obstaja, v odgovoru vrnemo kodo HTTP 404. Vsak klic v obliki niza skupaj z datumom in uro zabeležimo v podatkovno bazo. To je nujen podatek za odpravo morebitnih napak.

V naslednjem koraku avtoriziramo klic. Produkt smo v bazi že poiskali, zato poznamo vse njegove nastavitve, vključno s FastSpringovim zasebnim ključem. Kot že omenjeno, je vsebina klica v obliki slovarja. Slovar abecedno uredimo po ključu. Sedaj vse vrednosti zlepimo v en niz, le da varnostnega izvlečka ne pripnemo zraven. Namesto njega na koncu pripnemo zasebni ključ, ki smo ga predhodno shranili v bazi za izbrani produkt. Nad celotnim nizom zgeneriramo MD5 izvleček. Primerjamo dobljeni rezultat s prejetim varnostnim izvlečkom. Če se ta dva ujemata, smo plačilni servis FastSpring uspešno avtorizirali in lahko nadaljujemo z izdajo licence. V nasprotnem primeru je lahko posrednik v komunikaciji podatke spremenil, ali pa je zahtevo za izdajo licence sprožila tretja oseba.

Obliko podatkov ustrezno prilagodimo programskim funkcijam, ki služijo kreiranju naročila v partnerskem portalu in izdaji licence. Podatkov o stranki imamo na tem mestu bolj malo, a uporabimo vsaj to, kar imamo. Po drugi strani pa poznamo že vse nastavitve za izbrani produkt. Naj spomnimo, da smo le-te zapisali v podatkovno bazo, kar smo obdelali v poglavju o partnerskem portalu. V primeru, da se postopek uspešno izvede, FastSpringu vrnemo licenčno številko. V tem trenutku plačilni servis naročilo označi kot zaključeno, kar kupcu izpiše na ekran in mu pošlje kratko sporočilo o naročenih produktih.

Na tem mestu je potrebno poudariti, da avtomatizaciji pripisujemo visoko stopnjo poslovne vrednosti. Naš glavni cilj je, da stranka prejme licenčno številko takoj, ko je naročilo uspešno plačano. Ne želimo, da bi šlo ob kreiranju serijske številke kar koli narobe in bi stranka morala čakati nanjo, dokler ne bi nekdo od zaposlenih ročno uredil težavo. Poskrbeli smo, da se licenčna številka, v primeru, da je klic avtoriziran, brezpogojno izda. Če je prišlo v katerem izmed korakov do napake, se o tem z elektronskim sporočilom avtomatsko obvesti odgovorno osebo. Obvestilo vsebuje vse informacije o naročilu, ki so v danem trenutku na voljo. Stranka tako prejme licenco takoj, osebje v podjetju pa morebitne težave odpravi naknadno, a kljub temu, kar se da hitro.

### 6.4.3 Posodobitev podatkov o naročilu

Ob prvem klicu na naš API prejmemo zelo malo podatkov o stranki, saj želimo čimprej izdati licenco in ni potrebe po obdelovanju vseh podatkov. Sedaj, ko je naročilo zaključeno, imamo dovolj časa, da posodobimo podatke o celotnem naročilu v partnerskem portalu. Iz tega razloga sledi nov klic na naš API, tokrat na drugo končno točko. Na tem mestu obdelujemo že zaključena naročila. Tokrat se v HTTP POST klicu nahajajo praktično vsi podatki o naročilu v obliki JSON. Podatke najprej ustrezno serializiramo v objekt. Tudi ta klic avtoriziramo, saj ne želimo, da podatke o naročilu spreminja tretja oseba. V tem primeru smo prejeli en varnostni ključ za vsa naročila. Shranili smo ga v konfiguracijsko datoteko projekta, na mesto, ki je namenjeno shranjevanju vnaprej opredeljenih nizov. V glavi zahtevka se poleg ostalih nahajata še polji z varnostno vsebino in varnostnim izvlečkom. Na konec varnostne vsebine pripnemo varnostni ključ in nad celotnim nizom zgeneriramo MD5 izvleček. Primerjamo dobljeni rezultat s prejetim varnostnim izvlečkom. Če se vrednosti ujemata, smo plačilni servis ponovno uspešno avtorizirali in prejete podatke lahko uporabimo za posodobitev naročila.

Ustrezno naročilo poiščemo z referenčno številko, ki se ponovno nahaja v vsebini klica. Naročilu lahko sedaj nastavimo ustreznega „uporabnika partnerskega portala“, saj imamo podatek o tem, s katere spletne strani se je začel postopek naročila (mednarodne ali ameriške). V partnerski portal lahko zavedemo tudi, kolikšna je bila cena posameznega izdelka oziroma paketa v naročilu in, ali je stranka ob nakupu unovčila kupon za popust. Posodobimo še podatke o stranki, ki sedaj vključujejo celotno ime, naslov, naziv podjetja. Ko zahtevek obdelamo, FastSpringu vrnemo kodo HTTP 200, s čimer mu sporočimo, da smo zaključili z obdelavo naročila. V nasprotnem primeru mu vrnemo ustrezno kodo napake. Neuspešne klice lahko pregledujemo v FastSpringovi nadzorni plošči. Ko ugotovimo razlog za napako in le-to odpravimo, lahko klic sprožimo ponovno.

#### 6.4.4 Pošiljanje obvestilnega sporočila

Stranki se moramo zahvaliti za nakup in ji podati vse nadaljnje informacije. Že ob kreiranju produktov v partnerskem portalu smo izbrali, katera predloga elektronskega sporočila pripada določenemu produktu. Na tem mestu v bazi poiščemo ustrezno predlogo, v njej izpolnimo vsa polja o naročenem artiklu, veljavnosti licence ter o stranki in ji le-to pošljemo po elektronski pošti. S tem smo zaključili z obdelavo naročila.

### 6.5 Rezultat avtomatizacije

Želeli smo se prepričati, da smo z avtomatizacijo izdajanja licenc poslovni proces prenovili na bolje. S tem mislimo na izboljšanje uporabniške izkušnje strank in administratorjev partnerskega portala. Za kvantitativno analizo se moramo opreti na merljive rezultate. Osredotočili smo se na potreben čas za izvedbo naročila. Kot smo že ugotovili, prvi korak naročila, ki zajema vnos osebnih in plačilnih podatkov, ostaja praktično enak, zato smo čas merili od trenutka, ko je stranka oddala naročilo.

Prvotni sistem je zahteval posredovanje administrativnega delavca, ki je v partnerski portal vnesel podrobnosti o stranki in naročilu ter ji zgenerirano licenco posredoval na elektronski naslov. Tovrstno delo je v povprečju terjalo petnajst minut časa. Za kvalitetno primerjavo moramo upoštevati pogoj, da je administrator naročilo obdelal v trenutku, ko je bilo le-to poslano v obdelavo. Žal ta pogoj ni bil praktično nikoli izpolnjen. Administrator je bil vedno zaposlen še s kakšno drugo nalogo. Naročila, ki so bila oddana na ameriškem portalu, so prihajala ponoči, administrator pa jih je obdelal šele naslednji delovni dan. Prav tako je bilo z naročili, ki so bila oddana med vikendi in prazniki. Da bi oba sistema med sabo sploh lahko primerjali, moramo upoštevati tudi povprečen čas, ko je naročilo čakalo na obravnavo. Tega so v podjetju ocenili na dvanajst ur.

Sistem, ki smo ga razvili v okviru diplomske naloge, dosega neprimerljivo boljši čas. Od trenutka, ko stranka potrdi naročilo, pa do trenutka, ko na ele-

ktronski naslov prejme serijsko številko, ne mine niti minuta. Lahko bi dejali, da je čas zanemarljiv. Z veseljem ugotavljamo, je bila prenova poslovnega procesa uspešna in smo dosegli svoj namen.

# Poglavje 7

## Skrb za varnost

Glede na analize sedanjega stanja so uporabniki spletnega plačevanja podvrženi številnim tveganjem. Ponudniki spletnih plačilnih sistemov so ta tveganja analizirali in kot protiukrep ponudili številne rešitve. Še vedno se večina vdorov zgodi zaradi neustreznih gesel. Večina ljudi se poslužuje kratkih in znanih besed, ki jih zlahka najdemo v slovarjih in jih uporabimo za napade z grobo silo (angl. brute force). 86 % ljudi uporabi enako geslo oziroma omejen nabor gesel za prijavo v najrazličnejše portale in spletne storitve [19].

Z zlorabo kreditnih kartic se je pojavila potreba po standardu, ki bi se uporabljal kot glavna referenca pri zaščiti podatkov. Visa in MasterCard sta izdala mednarodni standard za zagotavljanje varnosti podatkov o kreditnih karticah, imenovan PCS DSS. Decembra 2013 je bila izdana zadnja verzija tega standarda, ki je prinesla veliko novosti in je povzročila precej pregledov tudi organizacijam, ki so že bile skladne s prejšnjo verzijo standarda. Občutljive informacije, do katerih vsiljivci lahko dostopajo, je potrebno med prenosom šifrirati. Napake v konfiguraciji brezžičnega omrežja in šibke točke v nastavitvah enkripcije in avtentikacije so lahko razlog za zlorabo podatkov [22].

Pri oblikovanju elektronskega plačilnega sistema je potrebno upoštevati ogromno dejavnikov. Dva izmed teh sta tudi učinkovitost sistema in var-

nost vsake transakcije. Iz tega razloga moramo zagotavljati zaupnost, avtentičnost, celovitost in nezatajljivost transakcij [12].

Metode za zagotavljanje varnosti lahko vključujejo prejemanje podatkov za vpis (prijavo) uporabnika in podatkov za varen dostop do aktivnih kreditnih kartic uporabnika. Ob tem morajo biti podatki za vpis uporabnika pravilni. Vključujejo lahko varno posredovanje podatkov o uporabniškem računu, uporabniški napravi ali registracijskemu strežniku. Na uporabnikovo zahtevo lahko metoda ugotovi, ali je na vnešenem plačilnem računu dovolj sredstev za izvedbo transakcije s podanim zneskom [11].

## 7.1 Zavarovana povezava HTTPS

API v svet izpostavljam preko zavarovane povezave po protokolu HTTPS. Na strežniku je nameščen SSL certifikat ponudnika GoDaddy [9]. Omenjena certifikatna agencija (CA) jamči, da so podatki v certifikatu resnični. Če zaupamo certifikatni agenciji, potem verjamemo podatkom v certifikatu. Ta vsebuje podatke o nosilcu in njegov javni ključ, podatke o certifikatni agenciji in podpis, obdobje veljavnosti ter dodatne informacije [5]. S certifikatom torej zagotavljamo svojo identiteto vsakomur, ki dostopa do našega API-ja.

Obstaja več protokolov, implementiranih za namen varnega e-poslovanja. Najbolj znan je gotovo protokol SSL. Transport Layer Security (TLS) in njegov predhodnik Secure Socket Layer (SSL) sta kriptografska protokola, ki zagotavljata varno komunikacijo preko interneta. Zelo pogosto se uporablja v elektronskih transakcijah. Protokol SSL je bil razvit v Netscape Communications Corporation. Protokol je sestavljen iz dveh plasti. Na najnižji ravni, razvit na vrhu nekaterih zanesljivih transportnih protokolov (npr. TCP), je protokol SSL, ki sprejme neinterpretirane podatke z višje plasti v nepraznih blokih arbitrarne velikosti. Rokovalni protokol, ki je sestavni del SSL-a, omogoča klientu in strežniku, da avtenticirata drug drugega in se dogovorita o šifrirnem algoritmu ter z njim povezanimi šifrirnimi ključi, predno aplikacijskih protokol prejema ali pošilja prvi niz podatkov. Velika prednost SSL-a



je, da je neodvisen od aplikacijskega protokola. SSL omogoča, da je promet med strežnikom in odjemalcem (spletnim brskalnikom) strogo šifriran z uporabo tehnologije javnih ključev. Slabost SSL-a je, da ne more preprečiti kraje osebnih podatkov [12].

Secure Electronic Transaction (SET) je bil osnovan s strani Vise in MasterCard-a v sodelovanju z različnimi tehnološkimi podjetji, kot sta IBM in Microsoft. Sistem zagotavlja, da številka kreditne kartice nikoli ne potuje po internetu v enem kosu, temveč je razdrobljena. Tako človek številke ne more zlahka razbrati. SET je torej eden izmed tistih, ki zagotavlja varnost v elektronskem poslovanju, temelji pa na digitalnih podpisih oziroma certifikatih (angl. Certificate Authority CA). Omogoča uporabo različnih kreditnih kartic. Služi za integracijo informacijskih sistemov, verifikacijo vseh finančnih podatkov in šifriranje občutljivih podatkov [12].

## 7.2 Enosmerna zgoščevalna funkcija MD5

Za avtorizacijo plačilnega servisa FastSpring uporabljamo različne zlepljenke nizov, vsem pa je skupno, da nad njimi uporabimo enosmerno zgoščevalno funkcijo MD5. To je kriptografska preslikava, ki preslika poljubno velike vhodne nize podatkov v nize točno določene dolžine – izvlečke. Iz izvlečka je težko izračunati prvotno sporočilo. Težko pomeni, da tovrstne operacije na determinističnemu turingovemu stroju ni mogoče izvesti v polinomskem času [5]. Ta način avtorizacije nam omogoča, da zasebnega ključa ne prenašamo po omrežju, hkrati pa zahtevamo poznavanje le-tega od obeh udeležencev v komunikaciji.

MD5 (angl. Message digest 5) je eden izmed zgoščevalnih algoritmov, ki ga je leta 1991 razvil Ron Rivest. Že leta 1993 so odkrili prve ranljivosti algoritma. Šibka točka le-tega so trki, kar pomeni, da je relativno enostavno poiskati dva različna vhoda, ki sta procesirana v enak izvleček. Omenjena pomanjkljivost je posledica napačne zasnove samega algoritma [21]. Leta 1993 sta Den Boer in Bosselaers našla prvo „psevdo-kolizijo”. Leta 2004 so

v okviru projekta MD5CRK, Xiaoyun Wang, Dengguo Feng, Xuejia Lai in Hongbo Yu pokazali, da je napad na MD5 mogoče izvesti v eni uri (na IBM p690 računalniški gruči). Leta 2006 je Vlastimil Klima objavil algoritem, ki je zmogel poiskati kolizijo na prenosniku v eni minuti. Danes so znani napadi, ki omogočajo izračun kolizije na podlagi dveh poljubnih nizov vhodnih podatkov v nekaj urah. Rešitev problema je uporaba boljše zgoščevalne funkcije, npr. SHA2 (SHA-256 in SHA-512 ter SHA-224 in SHA-384). Pri tej kolizije še niso našli, a je izračun počasnejši kot pri MD5 [16].

Pri izbiri zgoščevalne funkcije nimamo izbire, ker tak način komunikacije narekuje plačilni servis. Problem prehoda iz MD5 na drugo zgoščevalno funkcijo je ta, da gesel ne moremo dekodirati in ponovno zakodirati z novim algoritmom. V našem primeru omenjene težave nimamo, saj se ustreznost izvlečkov preverja le v trenutku komunikacije, ni pa tega potrebno izvajati za nazaj. Tako bi lahko FastSpring nenadoma zamenjal zgoščevalno funkcijo, mi pa bi naš API le rahlo modificirali. Razlog, zakaj tega še ni storil, nam ostaja skrit, vendar smo na spremembo pripravljeni.

## Poglavje 8

### Sklepne ugotovitve

Le prilagodljiva in fleksibilna podjetja lahko sledijo stalnim spremembam in so zato učinkovita in poslovno uspešna. Bliskovit razvoj interneta zahteva prilagodljivost potrebam in zahtevam strank. Cilj diplomskega dela je bil integracija platforme za izdajo licenc s spletnim plačilnim sistemom. Ključna težava obstoječega poslovnega procesa je bila čakanje stranke na posredovanje administrativnega delavca v podjetju, da ji ročno izda licenčno številko. Rešitev smo videli v avtomatski komunikaciji obeh sistemov.

Najprej smo se posvetili raziskovanju in primerjalni študiji med ponudniki spletnih plačilnih sistemov. Neustrezna izbira bi lahko podaljšala čas integracije ali celo pripeljala do spodletelega projekta, zato smo temu delu namenili veliko časa. Izbira najustrežnejšega plačilnega sistema je bil prvi korak do cilja. Z uporabo programa DEXi za večparametrično odločanje smo izbrali najprimernejšega kandidata.

V nadaljevanju smo predstavili uporabljene tehnologije. Potrebna je bila nadgradnja obstoječe platforme za izdajo licenc z razdelkom, v katerem lahko preko grafičnega uporabniškega vmesnika vnaprej definiramo prodajne produkte. Pri nadgradnji smo želeli slediti dobri programerski praksi in upoštevati obstoječo arhitekturo, zato je bilo potrebno dobro poznavanje platforme.

Ključna naloga je bila razvoj REST API-ja, preko katerega bosta sistema

komunicirala in si izmenjevala podatke. To so podatki o stranki, naročeni programski opremi in o licenčnih ključih. Na tem mestu smo morali poskrbeti za ustrezno varnost in zaščito podatkov, zato je API dosegljiv zgolj po zavarovani povezavi. Za dostop do platforme za izdajo licenc je potrebno poznavanje zasebnega ključa.

Prototip rešitve smo uporabili za testiranje v praksi. Izkazalo se je, da nas je prototipiranje pripeljalo do učinkovite rešitve. Iz rezultatov testiranja je razvidno, da smo uporabniško izkušnjo pri nakupu programske opreme močno izboljšali. Proces izdaje licence je po novem popolnoma avtomatiziran in ne dopušča vnosnih napak. Administratorji v podjetju so tako razbremenjeni rutinskega vnašanja podatkov v partnerski portal, stranke pa lahko začnejo z uporabo programske opreme že v nekaj trenutkih po nakupu. Skozi celoten razvoj smo stremeli k izboljšanju uporabniške izkušnje, kar nam je po naši oceni tudi uspelo.





# Slike

2.1	Postopek spletne transakcije . . . . .	12
3.1	Končne ocene kandidatov . . . . .	22
3.2	Graf ocene kriterijev za posamezne kandidate . . . . .	24
4.1	Visual Studio . . . . .	30
4.2	SQL Server Management Studio . . . . .	32
5.1	Kreiranje produkta v partnerskem portalu . . . . .	36
5.2	Seznam spletnih naročil v partnerskem portalu . . . . .	39
6.1	Diagram prvotnega poslovnega procesa . . . . .	43
6.2	Diagram novega poslovnega procesa . . . . .	44
6.3	Nadzorna plošča plačilnega servisa za urejanje in dodajanje produktov . . . . .	46
6.4	Določanje cen in pogojev posameznim produktom . . . . .	46
6.5	Nastavitve za oddaljeno generiranje serijskih številčk . . . . .	47
6.6	Naročilo - vnos osebnih podatkov . . . . .	48
6.7	Izbira plačilne metode in vnos plačilnih podatkov . . . . .	49
6.8	Vnos davčne številke in promocijske kode . . . . .	49
6.9	Obvestilo o zaključenem naročilu . . . . .	50





# Tabele

3.1	Odločitvena pravila za atribut ponujane funkcionalnosti . . . .	19
3.2	Odločitvena pravila za korenski atribut . . . . .	20
3.3	Primerjava plačilnih servisov . . . . .	21
3.4	Rezultati vrednotenja kandidatov . . . . .	23
3.5	Primerjava ponudnikov preverjanja davčne številke . . . . .	27



# Literatura

- [1] *Kratek priročnik jezika C# in razlike z jezikom C++*. Fakulteta za elektrotehniko, računalništvo in informatiko, Inštitut za informatiko, Maribor, 2008.
- [2] “Collection of cash items and returned checks,” [https://www.frbservices.org/files/regulations/pdf/operating\\_circular\\_3\\_07232015.pdf](https://www.frbservices.org/files/regulations/pdf/operating_circular_3_07232015.pdf), Jul. 23 2015, [Dostopano: 7. junij 2016].
- [3] W. Apricot, “Understanding online payment services,” <http://www.wildapricot.com/articles/understanding-online-payment-services>, 2016, [Dostopano: 5. marec 2016].
- [4] M. Bohanec, “Dexi: Program for multi-attribute decision making user’s manual,” *Ljubljana, Slovenia: Institut Jozef Stefan*, 2008.
- [5] J. David, “Elektronsko poslovanje, prosojnice z laboratorijskih vaj,” 2015.
- [6] K. Diveley in K. Hansen, “Payment service method and system,” Jan. 16 2007, uS Patent 7,165,052. Dostopno na: <https://www.google.com/patents/US7165052>
- [7] R. T. Fielding, “Architectural styles and the design of network-based software architectures,” University of California, Irvine, Dissertation, 2000.

- 
- [8] “Getting started with asp.net web api,” <http://www.asp.net/web-api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api>, [Dostopano: 7. junij 2016].
- [9] “Godaddy,” <https://uk.godaddy.com/>, [Dostopano: 7. junij 2016].
- [10] K. Jaklitsch, “Analiza spletnega nakupovanja pri slovincih,” Ekonomska fakulteta , Univerza v Ljubljani, Diplomaska naloga, 2007.
- [11] S. Kasower, “Method and system for secure online payments,” Jan. 22 2015, uS Patent App. 14/506,547. Dostopno na: <https://www.google.com/patents/US20150026040>
- [12] D. Khan, P. Varshney, in M. A. Qadeer, “E-commerce: From shopping carts to credit cards,” in *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, May 2011, str. 81–85.
- [13] S. Kumari in S. K. Rath, “Performance comparison of soap and rest based web services for enterprise application integration,” in *Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on*, Aug 2015, str. 1656–1660.
- [14] L. Li in W. Chou, “Design and describe rest api without violating rest: A petri net based approach,” in *Web Services (ICWS), 2011 IEEE International Conference on*, July 2011, str. 508–515.
- [15] Maestro, “Pogosta vprašanja,” [http://www.maestrocard.com/si/about/about\\_faqs.html](http://www.maestrocard.com/si/about/about_faqs.html), 2016, [Dostopano: 5. marec 2016].
- [16] “Md5 na zatožni klopi,” [http://matej.owca.info/predavanja/Md5\\_na\\_zatozni\\_klopi.pdf](http://matej.owca.info/predavanja/Md5_na_zatozni_klopi.pdf), [Dostopano: 7. junij 2016].
- [17] M. Mohamadi in T. Ranjbaran, “Effective factors on the success or failure of the online payment systems, focusing on human factors,” in *e-*

*Commerce in Developing Countries: With Focus on e-Security (ECDC), 2013 7th International Conference on*, April 2013, str. 1–12.

- [18] H. Olliphant, F. Chastagnol, Y. Su, W. Wu, C. Brown, T. Dang, T. Veino, C. Gunby, P. Chu, in S. Chen, “Method and system for facilitating online payments based on an established payment agreement,” Aug. 23 2012, uS Patent App. 13/458,855. Dostopno na: <https://www.google.com/patents/US20120215697>
- [19] Y. p. Cai in Y. y. Wang, “Simple said about online payment risks and preventive measure,” in *Information Systems for Crisis Response and Management (ISCRAM), 2011 International Conference on*, Nov 2011, str. 243–246.
- [20] Z. potrošnikov Slovenije, “Osebne finance - slovar,” <https://www.zps.si/osebne-finance/slovar/index.php?Itemid=678>, 2013, [Dostopano: 13. avgust 2013].
- [21] A. A. P. Ratna, P. D. Purnamasari, A. Shaugi, in M. Salman, “Analysis and comparison of md5 and sha-1 algorithm implementation in simple-o authentication based security system,” in *QiR (Quality in Research), 2013 International Conference on*, June 2013, str. 99–104.
- [22] M. R. Shihab in F. Misdianti, “Moving towards pci dss 3.0 compliance: A case study of credit card data security audit in an online payment company,” in *Advanced Computer Science and Information Systems (ICACSIS), 2014 International Conference on*, Oct 2014, str. 151–156.
- [23] “Tehnologije, ki jih uporabljamo: Asp.net mvc,” <http://www.avera.si/tehnologija/asp-net-mvc>, [Dostopano: 7. junij 2016].
- [24] “Tehnologije, ki jih uporabljamo: Microsoft sql,” <http://www.avera.si/tehnologija/microsoft-sql>, [Dostopano: 7. junij 2016].

- [25] P. Trusiewicz, M. Witan, in M. Kuzia, “Mobile payment system - telco 2.0 application dedicated for payments,” in *Computer Science and Information Systems (FedCSIS), 2013 Federated Conference on*, Sept 2013, str. 859–864.
- [26] B. P. Upadhyaya, “Rest client pattern,” in *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, June 2014, str. 231–235.